A Deep-learning approach to predicting availability of Industrial Gas Turbines

MSc By-Research

Lincoln University

School of Computer Science

MLearn Group

Jason Kenneth McGinty

October 2019

Abstract

In this thesis a new system is proposed which applies the techniques developed in the domain of Deep Learning to predict the ongoing availability of Industrial Gas Turbines (IGT's) in the range of 5 to 15MW. Deep learning has been targeted due to the recent rapid advances made due to the availability of increasingly mature software platforms, development methodologies and high-performance hardware.

The complexity of problem is explored in the literature review (such as the difficulty of modelling accurately the many different physical processes, noisy or missing data and the complex interdependence of many of the systems), and it is summarised that the focus of this work is one which does not appear to have been comprehensively tackled within the published literature (whose main focus has been on individual component reliability rather than the IGT in its entirety).

This thesis also demonstrates some of the progress that have taken place in recent years with regards the core algorithms and frameworks of deep learning by revisiting previous work and clearly showing the improvements advances such as Batch Normalisation and Adaptive Momentum (Adam) has made. Further investigation into the most appropriate such algorithms for this particular problem is also covered.

The proposed deep learning architecture uses a number of traditional network components (such as Multi-Layer Perceptron's, Convolutional Neural Network's etc.) in a way to solve a novel problem of overall IGT (and which is hoped is applicable in other complicated industrial environments) availability prediction problem and to account for the large gaps in visibility of issues that current systems encounter when attempting to predict availability for IGT's as single holistic system.

A further unique fact of this work is that it has also leveraged large amounts of data made available by a major OEM in order to realise the full benefit of a deep learning approach to this problem. This is in contrast to the majority of papers published to date which focus solely on a handful of units. This has made possible for the first time a digital twin wrought out of operation of an entire fleet of a diverse range of IGT's and allow for a comprehensive coverage of the factors pertaining to the maintenance of a high availability of the industrial assets targeted.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Industrial Gas Turbines or IGT's (as opposed to their more well known cousins Jet Engines, which although similar physically, have a substantially different operating profile) since their introduction in the late 1930's [ALSTOM, 2007] have often been found to be critical pieces of infrastructure for their owners who have invested considerable sums in their installation, maintenance and operation. Whilst expensive they have some unique features compared to other power sources such as low moving part count, high reliability, compact nature and fuel flexibility.

Because of the with the significant complexity and costs (the market just for the servicing of gas turbines for power generation and oil & gas use globally is expected to be $41.6 billion by 2025 [Grandview Research, 2018]) owners of such equipment expend, there are typically highly dependant on the equipment being reliable and running for a significant number of hours in order to receive a reasonable payback on their investment [Grosshauser, 2016]. IGT's are often integral to the processes that they are trying to support (such as pipeline transportation or production processes at a factory). Any issue with reliability can adversely affect production of these faculties and cause a significant setback in relation to the sites profitability and production schedule (as quoted in [Muhammad et al., 2017] a single LNG plant can be losing $25 million, or an oil rig approximately $7 million in lost production per day, in addition to the added safety issues of losing power on an off shore environment).

Due to the integral and often critical nature of the role that IGT's play in their operators businesses, it is often paramount that they have the turbines available as much as possible, or know well enough in advance that an issue is occurring that they can operationally deal with any such outage. This allows for the appropriate planning and scheduling of maintenance activities, not just upon the IGT but on ancillary plant equipment also, thus potentially minimising any disruption to the operations of the site.

Even just focusing on the repair and return to service of the IGT itself, it can be seen in the extensive description of the cost of involved between repair and replacement outcomes by [Jack, 2015] by having sufficient time to plan a repair you can substantially reduce the overall cost of the work required, by being able to undertake the optimum solution whether you are seeking to minimise downtime or capital costs) to return an IGT back into service. This trade-off is exacerbated by the long lead time for many of the specialised parts that are utilised by IGT's which may result in limiting OEM's and operators options for a timely return to service where an event is unexpected. When a serious issue occurring on an engine (such as vibration spikes being noticed but not acted upon) being left to chance or the last minute before being dealt with, the opportunity to utilise the most appropriate method may be further constrained by factors such as parts or manpower availability.

Also without the foresight of up and coming events that could disrupt the availability of the site the scheduling of other equipment that is not currently the focus of the immediate problem tends to get overlooked in the effort to return the equipment to service, this in itself can lead to further downtime as other scheduled downtime that could have been avoided by being brought forward is often infeasible if replanning is left till the last moment due to constraints on other suppliers. This ultimately leads to a bad outcome as regards availability for the customer and is a further indirect effect that can lead to unnecessary downtime and further losses of the magnitude identified earlier.

1.1 Overview of IGT's

IGT's and are very similar to the early forms of Jet Engines that were common in the 50'ts and 60's and are typical described as either single or twin shaft (i.e have two shafts running in line along a single axis) as opposed to the multiple spool engines that are common with Aero-derivative (those engines based on existing jet engines but modified for use in an industrial setting) engines (there are significant overlaps in either engines capability). Pure IGT's are designed for simpler maintenance and long sustain running periods. Aero-derivative's are typically focused at markets where short term peak performance is needed (such as a "Peaking" plant to handle short term intermittent loads upon a grid) whereas IGT's typically focus on long-term base load operation.

Due to their more rugged and serviceable (such that most operations can be undertaken at site rather then a specialised factory if necessary) design IGT's are

Figure 1.1: Outside rendering of an IGT gas turbine package. Copyright Siemens AG - Used with Permission

typically preferred in remote and isolated environments due to their generally high reliability.

1.2 Aim

The aim of this work is to make progress in the field of predicting the short to medium term (i.e upto 30 days) availability of an individual IGT in a way that allows sufficient notice that other events can be scheduled appropriately around such issues and allowing for the customer to maximise operational uptime for its entire site where possible.

1.3 Objectives

Bearing in mind the issues identified above, it has become apparent that the system in the first instance should operate on the basis of how to

- sufficient data pertaining to individual engines should be extracted and prepared
- previous work should be replicated using accessible data to assess progress in available methods
- a model for predicting high level configuration of engines should be produced
- a model for predicting the data (Tags available) configuration of the engine should be produced
- Develop a generalised model of a non-specific range IGT (i.e targeting more then a single model but a whole family of physically different design of IGT).
- Develop an availability prediction model for IGT's which utilises the above generalised model.
- Identify potential future enhancements and work that could expand upon this bases.

Chapter 2

Principles of Gas Turbine Operation and Maintenance

2.1 Operation of a Gas Turbine

Gas Turbines operate on what is known as the Brayton cycle, this is very similar to that used in the commonly encountered 4-Stroke petrol engines, and can be summarised as "Suck, Squeeze, Bang and Blow". In a Gas Turbine however these activities effectively all happen continuously whilst the the engine is running, rather then as discrete steps as in the 4-Stroke engine. As can be seen in Figure 2.1, air is sucked in by the compressor and as it travels axially [1] along the compressor it is slowly squeezed. The air is then moved to the combustion chamber where fuel is added and ignited which leads to an expansion of the combusted air which is squeezed

[1] Radial turbines do exist and essentially follow the same process, but are not relevant within the context of this work as they typically are only used for very small scale instillations of less then 1MW.

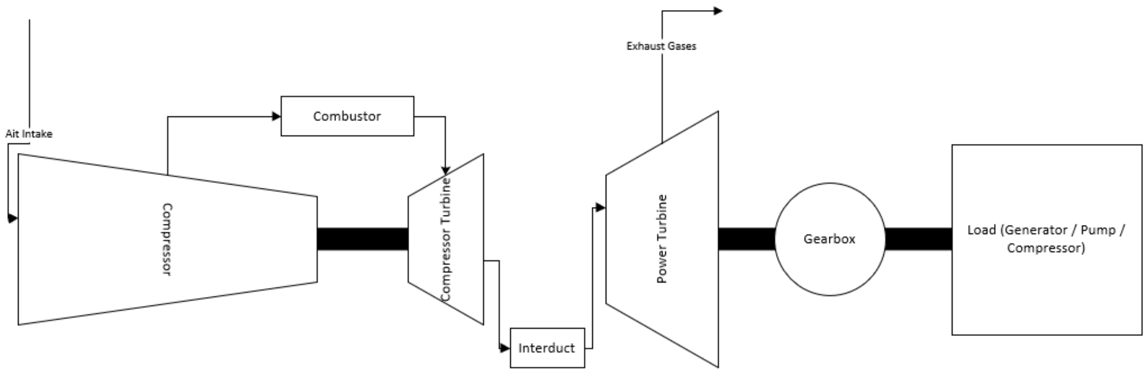

Figure 2.1: Simplified overview of the internal arrangement of typical IGT

out towards what is known as the compressor turbine. It is the job of the compressor turbine to capture enough air in order to continue to drive the compressor and suck more air in, allowing the engine to self-sustain. Useful energy is the captured by what is known as the Power Turbine. This device sits in the flow of air and again takes energy from the expansion of the gas moving past it and turns that back into rotational movement which can be used to drive equipment such as pumps, compressors and generators. Excess heat that is not used to either drive the internal compresor of the IGT or the Power Turbine is often used then to provide heat to a boiler (or occasionally used directly in the process such as in Medium Density Fibreboard production) and so provides an energy process with a high overall thermal efficiency compared to diesel engines or other fossil based energy sources.

2.2 Inherent Difficulty in the Problem

Turbines are difficult pieces of equipment to model accurately as there are a number of areas where there is limited direct operational visibility of the engines operating characteristics such as fuel quality and accurate measures of the degradation of the

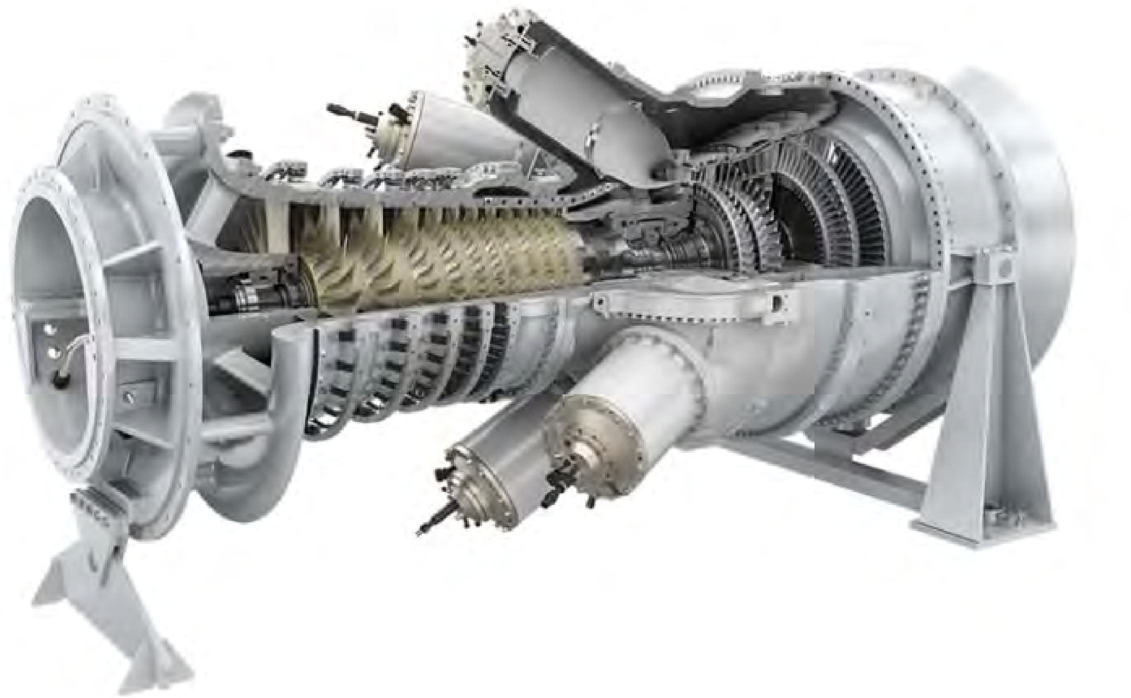

Figure 2.2: Exposed Cut-away view of an SGT-400 Industrial Gas Turbine. Copyright Siemens AG - Used with Permission

components. Degradation is also a significant issue, and os variable with each site's location (which accounts for differences in contamination and ambient conditions) and operating parameters (such as performance levels).

Further the complexity of the IGT as a complete system including all its supporting sub systems such as fuel regulation, lubrication, ventilation and torque transmission (such as gearbox or torque converters), rapidly increase the scope of any such predictive maintenance system. These difficulties were recognised and described by Tiddens et al. [Tiddens et al., 2018], where a structure was proposed for classifying by risk and effect the different sub-systems so that focus could be directed upon the leading causes, and from reviewing the literature in the next chapter it is clear that this is the approach that has traditionally been taken by researchers, to coalesce over the perceived higher value failures and exert significant effort in trying to solve each of these individual issues. The purpose of the work contained here-in differentiates itself from that individual approach and attempts to investigate the IGT as a holistic package, and compare it to the whole fleet as its ultimate distribution of failure rates.

The key categories that cover the vast majority of the issues within this domain can be defined as the following :

- Health Degradation Trends
- Defining the scope of any given fault with limited examples
- Low-dimensions of visualisation for highly complex data
- Non-linearity with respect to operating conditions
- Complexity of the interconnected physical processes

- Interdependence of the many systems with limited viability

2.3 Technical Challenges

2.3.1 Product variation

The manufacture of a gas turbine is still more of an artisanal pursuit then that of the say the manufacture of a car engine, where the tolerances and manufacturing capabilities are well known and remain constant throughout their life. Due to the lower volume and the complexity of the parts, there is a larger degree of individuality between the various units that are built. Many parts are hand fitted using traditional fitting techniques. It should also be noted that the majority of IGT's in service will have a combination of new and old overhauled parts which adds to the uniqueness of each unit and the complexity of understanding the interrelationships between the components.

2.3.2 Degradation

There are a number of degradation modes that occurs within a gas turbine over time, these are described in general as applied to gas turbines by Kurz [Kurz et al., 2009] and more specifically in relation to their on IGT's in Cruz-Manzo [Cruz-Manzo et al., 2018], in summary these include :

- Fouling : such as the ingress of dirt onto the compressor rotor leading to disrupted airflow and loss of efficiency
- Corrosion : caused by the ingress of volatile constitutes within the fuel or

Figure 2.3: Example of an SGT engine being built in the factory (rotor installation stage). Copyright Siemens AG - Used with Permission

inlet air. This can have the effect of causing damage to the high value metals contained within the engine.

- Erosion : This is typical caused by small particles escaping the filter and abrading away at the turbine components
- Fatigue : As the engines age they will suffer cyclic fatigue
- Creep : another function of the ageing and use of the machine is creep, where the structure of the metals subtly changes over time due to the nature of how a gas turbine operates.

It has also been highlighted in [Kiakojoori and Khorasani, 2016] the issue of concurancy of the various degradation mechanisms present within an IGT and where there is a interference between the signals you would expect to see from them individually occurring. A prime example cited is the ongoing operationally induced fouling of the compressor (of which some performance may be recovered through washing) but the ongoing erosion or corroding of the turbine and thus the overall reduction in IGT performance (as measured by efficiency or maximum output depending on the customers viewpoint).

The first of these issues (fouling) can have their effects minimised by the use of the of specialist procedures such as washing the compressor, or improved filtration, however no such method will completely eliminates these modes of performance degradation which are effectively a hazard of operation but will affect different sites and different engines in subtly different ways depending on the causative factors (such as poor quality air or fuel etc). As [Almasi, 2016] states "accumulated dirt in the air-compressor section of a gas turbine can result in increased fuel consumption more

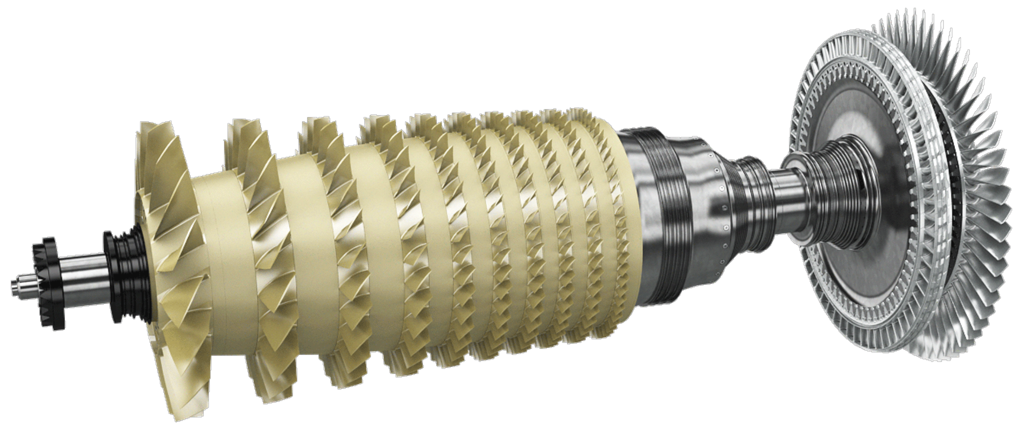

Figure 2.4: Model of the rotor of an SGT-400 IGT. The degradation described above largely affects the components on this element of the gas turbine, with contamination and corrosion building up on the blades surface erosion mechanisms removing material from them and finally fatigue and creep affecting their mechanical properties

frequent maintenance outages and decreased hot section life" and much of the information needed to predict that should be discernible from inspection of a the IGT's performance at a specfic load point.

Items which can have an effect on these are running profile (does it load up quickly or slowly). Many of these circumstances can lead to an early failure for an engine. This in effect means that IGT's build up a memory of their operation through the rats of degradation that they will suffer through there own individual operation patterns and environments in which they are situated. This in itself posses and interesting issue in relation to the prediction of availability of an IGT, as you would ideally wish to be able to assess where it is along the degradation path, even with imperfect knowledge of all its previous operational statistics. There are many approaches to trying to build health indices for prognostics and these are described in the next section.

2.3.3 Compressor and non-linearity

One difficulty with the Brayton cycle is that many of the various thermodynamic processes that occur during it are non-linear in their responses to changes in operating condition. This is highlighted the visualisation of the processes where the different parameters have a clearly non-linear relationship, which also varies further based on climatic conditions such as temperature, humidity and altitude and the types of fuels used (all of which have varying effects on the densities of the working medium within the engine).

The compressor of a gas turbine operates in a non-linear fashion, which has made it somewhat difficult to model accurately at any point other then its design point (i.e

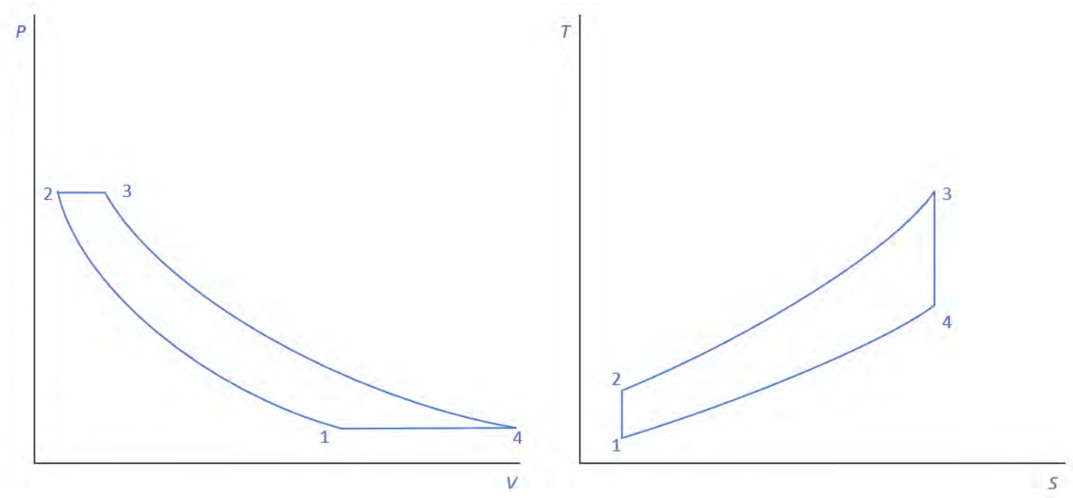

Figure 2.5: The Brayton Cycle, P = Pressure, V = Volume, T = Temperature, S = Speed, The points on (1, 2, 3, 4) correspond to the Suck, Squeeze, Bang, Blow process described in the opening of this chapter

full load).

2.3.4 Combustion System

The health and efficient operation of the combustion system is of crucial importance to the Gas Turbine, especially as the more sophisticated techniques are being brought to bare upon the issues of NOx and CO2 production. These more sophisticated system also become more sensitive to issues such as fuel quality and contamination that may not have been a problem with simpler less efficient and less polluting combustion system.

Issues occurring with combustion (such as lack of cooling iar, combustion instability or contaminations such as oil carry over) can result in material failures of parts which subsequently can damage other areas of the gas turbine such as compressors or turbines etc.

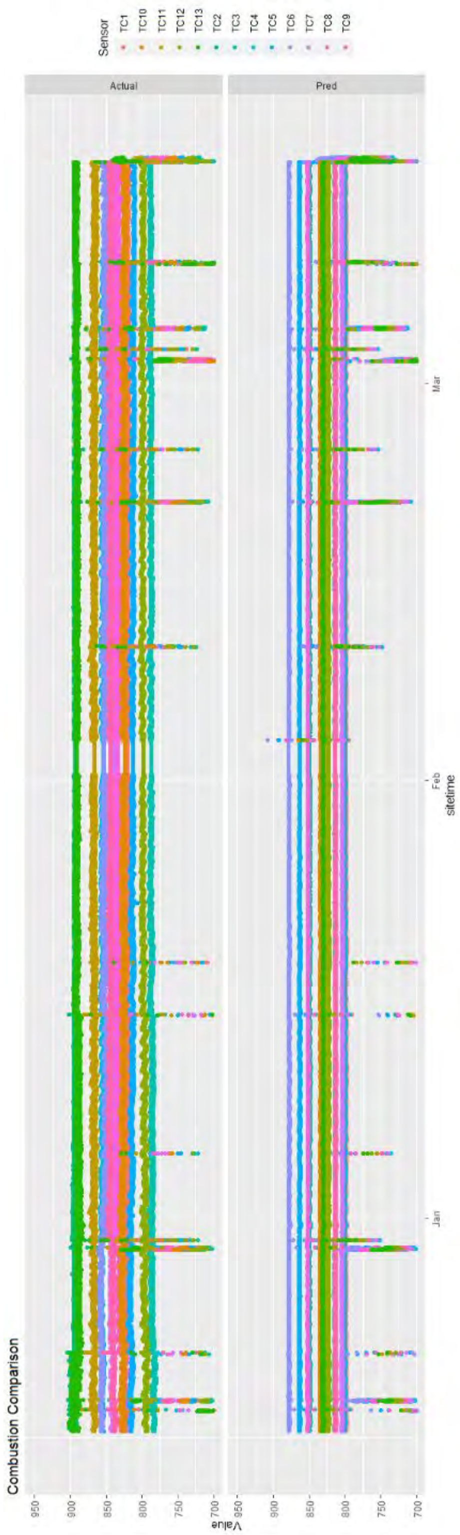

Figure 2.6: Example of combustion system and a generalised prediction from a basic digital twin model

2.3.5 Vibration

Vibrational measurements are one of the most common form of health measurement used in the diagnosis and prognosis of major issues within a gas turbine. There are a variety of ways in which the vibration reading can be recorded such as direct displacement or acoustic methods (such as accelerometers or acoustic emissions sensors). The variety of measurements reading the same value but operating in different ways can sometimes make it difficult to correlate the results between them confidently.

In the current case the majority of vibration sensors recorded are taken by way of direct displacement though a small number do read the acoustic emissions through the use of a accelerometer.

2.3.6 Ancillary Items

Many of the causes for IGT unavailability come not from the core engine but from there being an issue with the supporting equipment such as gearboxes, pump, valves (or their associated actuating devices) etc. There is a large amount of variability in how these are selected dependant on the preference of what are known as the packagers and the operational requirements of the customers. An IGT installation (as shown in Figure 1.1) us often refered to as a package. Packagers are the system integrators who build the "Package's" in which the IGT's are located and operated from. Many OEM's offer a packaging service themselves but many such engines are packaged by Third-Parties who will specialise in adapting IGT's for a specific market (such as off-shore or Combined Heat and Power (C.H.P)).

2.4 Limitations

There are a number of limitations as to what can foreseeable be predicted using the methods anticipated in this work. These are summarised in Venturini [Venturini and Puggina, 2012] as being :

- Foreign Object Damage : such as debris from ductwork impacting on the compressor rotor
- Unforeseen Site Issues : events can occur at the location of the gas turbine that are invisible to the turbine data collection processes and which can result in some degree of damage or effect. Such as gas or compressed air availability being restricted during operation.
- Hidden elements : such as pipe fretting etc, which will be impossible to identify on a given any given package with the information currently available within the scope of this work.

2.5 Maintenance Life cycle of an IGT

The expected life of an IGT varies depending on the product itself. Typical life-cycles can very from 24,000 hours Minor Overhaul (where significant number of turbine and combustion parts are replaced) / 48,000 hour Major Exchanges (whereby the engines are changed in their entirety). This split maintenance cycle adds and extra layer of complexity as noted in [Zaidan et al., 2015] that the profile of degradation can change significantly due to normal maintenance. This is quite an issue when there are approximately 20 levels of scope (such as simply changing burners, to replacement of all gas path parts including blades and discs etc) that could be involved in the regular maintenance of engine. Many sites due to local operating activities will require unique regimes on certain components to deal with issues such as fuel contamination (which can affection combustion equipments performance) or poor filtration(which can lead to errors ion of compressor and reduced performance). In between these major / minor overhauls of the prime mover of the IGT the ancillary equipment such as gearboxes and valves will need to be maintained. This can have a wide variety of scales but will often be tied in where feasible with the larger outages or on a annualised cycle. The 48,000 hours major interval period identified above is increasingly become more flexible as the operating loads of IGT's are being varied (such as reduction operation to minimise emissions rather then export surplus generation capacity) which can lead in some circumstances to extension of life of components that are lifted (which effectively means it should not be used any more in an operational engine) based on the equivalent stress they have suffered at different loads (such as 50 or 100% of maximum electrical output) or cyclic regimes (such as starting and stooping everyday rather then continuous operation).

From a commercial point of view OEM's offer a number options in relation to managing the servicing of IGT's, from simply providing parts to customers to fully managed service agreements which take into account future planned and potential unplanned outages (almost in the form of a insurance policy).

These can be roughly sub-divided into three type of maintenance strategy as described [Diallo, 2010] :

- Run to failure : Where the equipment is operated to the point of breakdown and then fixed on demand

- Prescriptive (or standard preventative maintenance) : Equipment is systematically inspected and parts regularly replaced in light with best practice guidance.
- Condition based maintenance : This is where the equipment is continually monitored and maintenance is planned around the assessment of that monitoring.

Each one of the above items has drawbacks, for instance running to failure can lead to far more downtime, especially as equipment ages as further damage may be caused by the original failures, and at a site level this type of approach can lead to extensive knock on consequences (such as losses in manufactured products that might have been wasted due to the sudden shut-downs caused). With prescriptive maintenance it is often the case that equipment is replaced and discarded when it has significant useful life remaining. Potentially the most promising approach is the Condition based maintenance, however whilst great strides have been made in the ability to model IGT (and other gas turbines) health.

The running regime of the IGT can also have a significant effect on the maintenance of the IGT as there are various elements that can be lifed on either calendar age (such as gaskets and seals), hours run or starts incurred or some combination of them both. As can be seen from 2.7 There are a variety of modes of operations which can change through the life of the unit (such as the units with plateaus or very slow accumulation of hours where they are used as backup units).

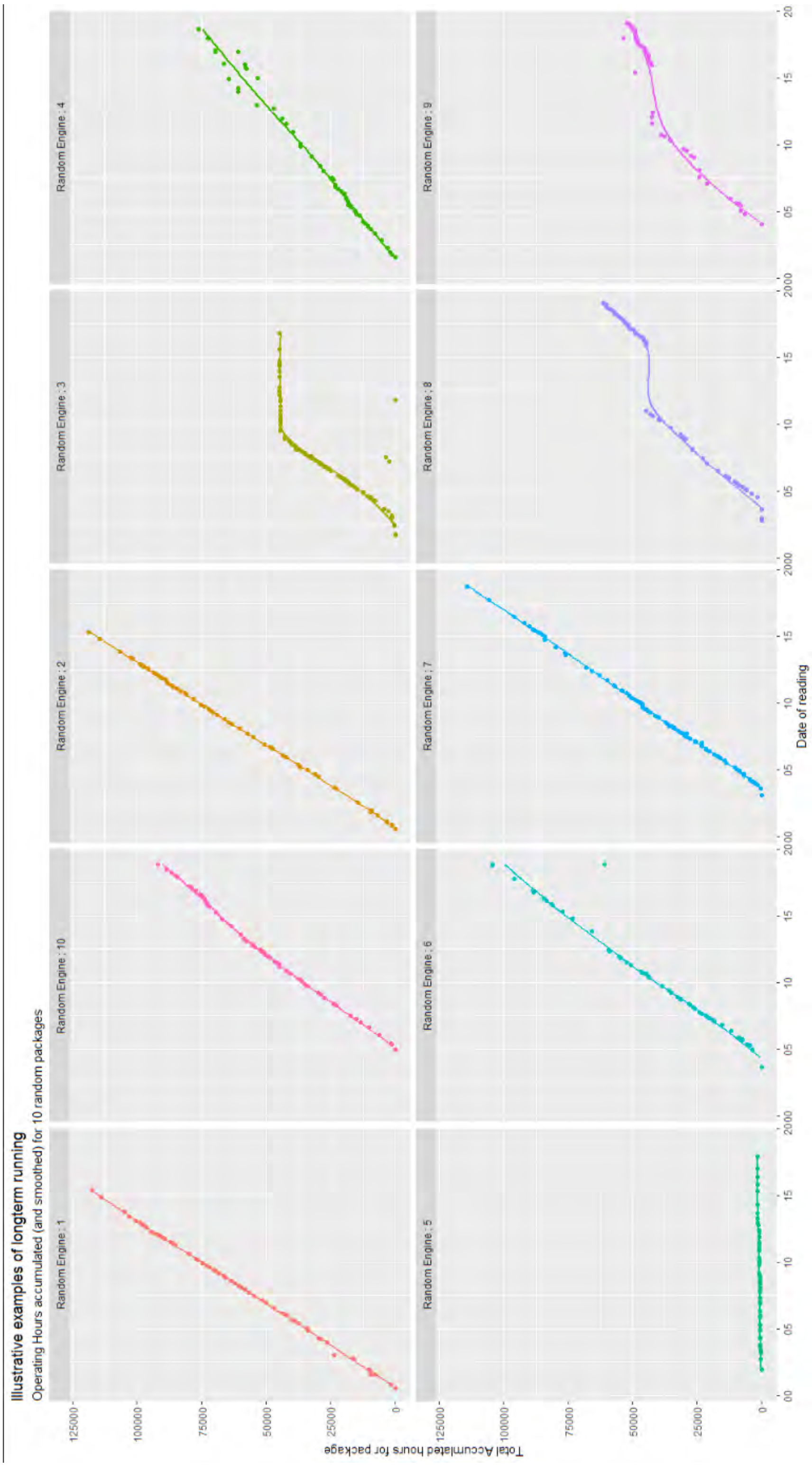

Figure 2.7: Individual faceted view of each engines running profile points of reading shown. Note the deviation that can occur, and a number of outliers that can exists due to errors in the collection process

Chapter 3

Review of Current Approaches in Deeplearning

3.1 Application of Deep-Learning

Deep learning is the latest moniker for what originally termed cybernetics or constructivist approaches that has been developing since the 1940's. The current iteration of the approach which has come to be known as "Deep Learning" as reviewed thoroughly in [Goodfellow et al., 2016] originated around the mid 2000's thanks to advances in both the algorithms available and the computation equipment needed to undertake the training and operation of the networks at ma much larger scale then had been previously envisioned. Whilst the overall technique is not new and has been regularly reviewed since the 1950's limitations always seem to prevent its movement into real-world applications.

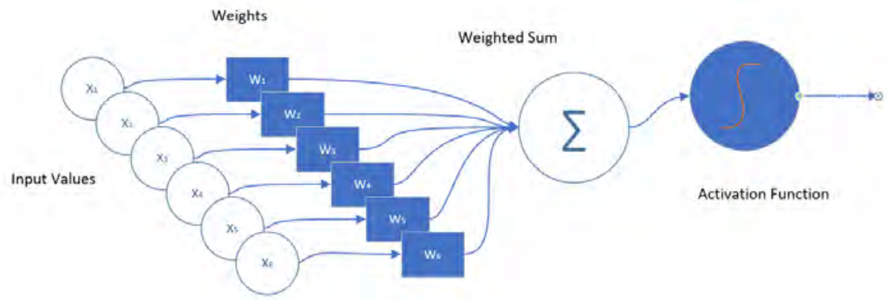

Figure 3.1: Outline of the structure of a single perceptron

3.2 Common Applicable Architectures

I do not anticipate covering every possible architecture that exists, but will primarily concentrate on those that have been explored or found useful within the context of this work.

3.2.1 Multi Layered Perceptron

The Multi-Layer Perceptron (or MLP) is an extension of the original Perceptron Hidden Layer feed forward network, which is stacked deeper and deeper (hence the name taken by the overall area of AI research). Whilst not commonly the focus of research on its own now due to more advance networks being utilised, the densely connected layers from which it is made up are still essential as the final layer in most other architectures.

Vanishing Gradient

On of the key issues that has had to be minimised is the issue of the vanishing gradient. As back-propagation increased the ability to train deeper and deeper networks, there became a limit in that beyond a few layers the updates were becoming so small they would be ineffective in allowing the network to train. This issue was overcome to some extent through the use onew activation functions (which are covered later) and now with even deeper networks the use of residual connection to allow for the changes to propagate more thoroughly through the network

3.2.2 Autoencoders

Autoencoders are another item that Geoffrey Hinton (who has been involved in a great many of the fundamental developments) popularised [Goodfellow et al., 2016, chap. "14]. They operate by having separate encoder and decoder segments which are then forced to build a common representation of the problem space but with a reduced dimensionality. This is visualised in 3.2. It has also been used susccessfuly in a number of IGT related works.

3.2.3 Convolutional Neural Networks : CNN

Convolutional networks have been instrumental in fostering the resurgence of interest in ANN's. They are originally inspired by the noble award winning work of Habel and Wiesel [Hubel, 1958; Hubel and Wiesel, 1959] in the late 1950's, whereby they were able to show the sensitivity of the different parts of the visual cortex to to different features, and that they were responsive to different components within their "receptive" fields such as horizontal or diagonal lines.

These were eventually put to good use by LeCun et. al. [LeCun et al., 1998] in the production of a handwritten digit classifier for US Zip Code addresses. he also introduced the convolutional building blocks that has today been largely standardised

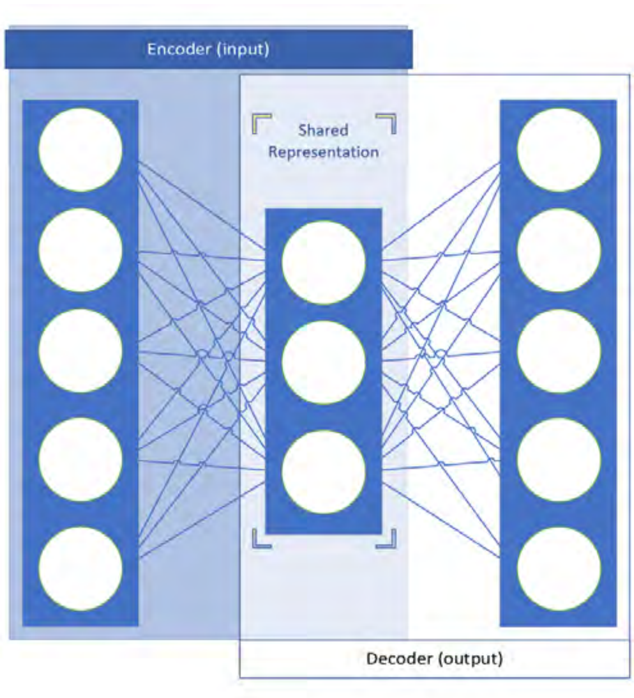

Figure 3.2: Block layout of an Autoenoder architecture

on such as Convolutional Layers and Max Pooling Layers.

Convolutional networks work in a slightly different way to the MLP described above in that rather then every layer is fully interconnected, a convolutional operation is used to reduce the amount of information absorbed by the network. These convolutional operations will result in a reduction of the amount of data available but will learn from the patters that emerge as a result of the convolution function. If we take an image of 28 pixel height and 28 pixels wide (such as those found in the infamous MNIST dataset produced by [LeCun et al., 1998]) for example and a 2D convolutional operation. the "convolutional window" (for illustrative purposes in this case instance 3 pixels wide) and with a "stride" (i.e. the distance it will move across the target windows with each step) of 2 will move across the top row of the image in 14 operations. Each time taking a value based on the 3x3 window (or filter) that it has viewed. This is a significant reduction in the number of trainable parameters and such layers get stacked it is noted that the deeper layers are learning some vary high level features.

Further in addition to the convolutional layers a CNN typically consists of pooling layers, which will average out and further down sample the filers. For instance when looking at images of cats. The lower level filters will be trained to identify elements such as horizontal or vertical lines. Higher level feature will eventual build upon these primitive filters to find combinations will actual represent higher level features such as eyes and ears for instance.

It is belief that these patters exist in the basic representation of engines operations being used in this project. Similar ideas have been explored on smaller scale by [Tayarani-Dathaie and Khorasani, 2015] time series prediction problems using a form

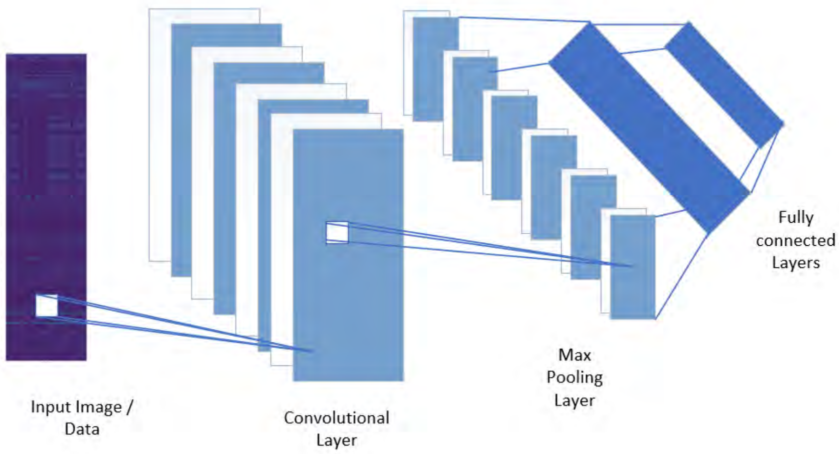

Figure 3.3: Example of simplified convolutional network architecture applied to the IGT dataset

of 1D Convolutonal network called a Time Delay Neural Network (where the time delay came from the striding operation being applied to timeseries data), these were applied to the concept to the identification and classification of faults on along the gas path of a aero derivative gas turbine.

Potential Trimming Strategies

As can be expected there is a significant increase in the number of trainable parameters that occur with this architecture, its possible that many of these early connections may become redundant as training progresses and the weights throughout the network are adjusted. Dropout whilst compatible with this architecture does not appear to be entirely [Feng et al., 2019]

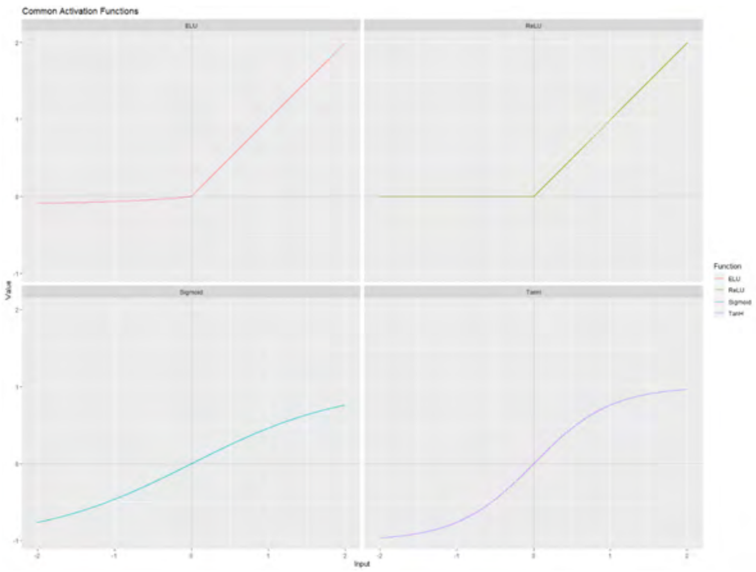

Figure 3.4: The four common activation functions used in this work visualised for comparison

3.3 Activation Functions

3.3.1 Sigmoid

The Sigmoid (as shown in 3.4) was one of the earliest (along with Tanh below) non-linear activation functions used in deep learning. It is still commonplace when being used for binary classifications. A computational light version known as the "Hard Sigmoid" is also now gaining use

$$f(x) = \frac{1}{1 + e^{-x}} \tag{3.1}$$

3.3.2 tanh

This was one of the original activation functions that is typical still in common place within the use of Recurrent Neural Networks (or RNN, this is a network commonly used in sequence processing), where is used as it remains more highly differentable with the sigmoid function that is used within the gate that identifies whether or not data will be able to pass back into the RNN.

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{3.2}$$

3.3.3 ReLU : Rectified Linear Unit

The ReLU which was first proporsed by [Hahnloser et al., 2000], and as with many areas within deep-learning has a biological analogue with the function of the neocortex of the brain. It has gone onto to become one of the most popular (except for some cases such as LSTM) activation function for Deep Learning, despite many attempts such as SWISH by[Ramachandran et al., 2017] o elu by [Clevert et al., 2016]. These have the promise of being a more effective activation function then even the ReLU, albeit it with a increased cost in computation, but so far have failed to make a significant impact.

It was noted in [LeCun et al., 2015] that the reason for the success of the ReLU activation function was due to it ability when used in the hidden layers of networks to improve the linear separability of the data as it was transported through the network to its final linear separation at the output layer.

The activation function can be easily represented as

$$f(x) = max(x, 0) \tag{3.3}$$

It operation is demonstrated in figure 3.4

3.4 Training

3.4.1 Metrics and Losses

As a result of the gradient descent operation it is necessary have a value to minimise, in this case the minimisation is undertaken by the use of what is known as a loss function. Many of these functions can also be beneficial as metrics to understand the performance of a given network.

Mean Squared Error and Root Mean Square Error

It tries to minimise the overall error regardless of whether it is positive or negative and is also somewhat resistant to individual outliers.

$$MSE = \frac{1}{x}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{3.4}$$

A further modification is the use of RMSE which has the advantage of being more comparable to the scale of the error actually present within the training data. As the two are closely related there are largely interchangeable.

$$RMSE = \sqrt{\frac{1}{x}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2} \tag{3.5}$$

R^2

The R^2 measure (which is also known as the coefficient of determination), it is an indicative measure of how well the output from a particular model fits with the original ground truth that it is trying to predict.

$$\text{Explained Variation} = \sum (Y_i - \hat{Y} + (i) \tag{3.6}$$

$$\text{Unxplained Variation} = \sum (Y_i - \bar{Y} + (i) \tag{3.7}$$

$$\mathrm{R}^2 = 1 - \frac{\text{Explained Variation}}{\text{Unexplained Variation}} \tag{3.8}$$

3.4.2 Optimisers

The optimiser is the algorithms that handles the training and manipulation of the weights to hit the anticipated target.

Stochastic Gradient Descent

For many years this was the primary gradient descent approach used in the training of Neural Networks and to be first presented in relation to a form that is easily encodable for machine learning applications to Kiefer & Wolfowitz [Kiefer and Wolfowitz, 1952]. Whist now somewhat less popular there is still evidence to suggest that it can perform as well if not better then the later algorithms below in relation to finding the optimum solution however it may require significantly more computational power to get to that level. For the purposes of this work it has largely been left unexplored expect in its use for replicating previous work undertaken, however there is a significant continuing interest in the use of this optimiser in production systems.

RMSProp

RMSProp was a further development that built on various ideas regarding the use of momentum of change of the gradient to self tune to finding the optimum solution. The idea was first presented in lectures notes by and has since been adopted as popular and efficient optimisation algorithm by Hinton [1]. It overcomes the problems encountered by other momentum based optimisers by slowing down the learning rate decay.

ADAM

The Adam algorithm (or adaptive momentum)was first proposed by Kingma & Ba in 2015[Kingma and Ba, 2015] and purports to integrates to some extent many of the benefits of both the previous AdaGrad and RMSProp. It quickly gained significant use in the years following its initial publication and as of 2017 [**?**] appears to be the most popular optimiser in use as referenced in publicly available research papers.

From a practical point of view Adam is easier to use as it automatically tunes its learning rate hyper-parameter, which is often managed by way of a learning rate scheduler (or other mechanism such as decay on plateau style

Recent research however has found that there are use cases where Adam provides a less then optimal solutions to the optimisation of neural networks despite its apparent popularity. This is because whilst Adam offers immense flexibility in relation to the capability to quickly converge on training data it appears that models optimised with

[1] Please refer to Neural Networks for Machine Learning : Lecture 6a Overview of mini-batch gradient descent for more information regarding RMSProp

tend to have lower overall generality then those optimised using RMSProp or SGD.

3.4.3 Training Tricks

A number of approaches can be scheduled using the Keras framework through its callback mechanism to tune hyper-parameters. As has been noted in in Goodfellow et al.[Goodfellow et al., 2016] there is a trade-off between the learning rate and the risk of falling into a local minima This can be offset to some degree by actively adapting the learning rate so that it slows down once learning appears to have stopped. This potentially reduces the penalty that a low learning rate would have on overall convergence.

Another useful approach is early stopping. This is where the training algorithm itself choices a time when a models training should be stopped (such as degredation in validation performance from best achieved). Due to the flexibility of high level frameworks such as Keras, it is possible to quite easily prototype and investigate novel scheduling regimes and other training techniques.

3.5 Pre-processing Data

Normalisation of Data for Deep-learning has been undertaken since the early days of ANN and the technique was popularised by LeCun [**?**]. The technique in general has proved beneficial in relation to minimising the difference and updates necessary to alter the weights held within the network. This leads to more rapid convergence of networks when training and generally more accuracy. There are a number of different approaches which have been summarised **?** such as Min/Max, Median, Sigmoid and the approach ultimately chosen for this work of using Z-Score standardisation.

Batch Normalisation

In addition to the preparation of the input data, there has been increasing research into the use of normalisation on the inputs to the individual layers. This has proven increasingly beneficial when dealing with extremely deep networks.

Dropout

Dropout was first proposed in [Srivastava et al., 2014]is a mechanism that is used to reduce the number of neurons required to reach a given results. It has the benefit of reducing the possibility of over fitting but does come at a significant computational cost during training. It also relatively easy to implement with little detrimental effect on complexity, training time, although some penallity is incurred in training accuracy in lieu of purported better generalisability.

3.6 Simple illustration of advances

Although it does not appear to be the case that a data driven model has been used directly in the production of availability estimates of the nature described here, there have been a number of attempts over the years to produce simplified ANNs that represent the relationships between the various processes within the IGT. To begin with it was felt that replicating with the available data some of the prior approaches that had been used for this approach. The first that was of relevance was that identified and previously discussed by [Fast et al., 2009a] and later thesis [**?**]. This was an early attempt at producing a version of a specific engine based on similar input parameters.

The network was what would be described now as follows :

- Multilayer Perceptron, which is effectively just a three layer (input, hidden, output) densely connected neural network
- Tanh activation function
- Stochastic Gradient Descent

The initialisation schema itself was not identified however at the time it was prevalent in most not specialised applications to zeros the weights.

It is also not clear exactly how many neurons are contained within the hidden layer, but in this case 10 was used (which appears to be the maximum referenced). This resulted in a reasonable number of trainable parameters as it was not anticipated any hardware acceleration would have been available at that time (as hardware acceleration libraries such as cuDNN do not appear to have been released or commonly used until circa 2015.

The data was analogous to but not exactly the same as that which was used for the test. A different model of Gas Turbine was used, although taken from a very similar family of Industrial Gas Turbine (as opposed to an Aero Derived Industrial Gas Turbine) both being ”Twin Shaft” designs, and similar parameters were chosen.

In terms of input data 4 months of data were extracted for a single engine. It was filtered to only contain data when the engine was ”on load” of producing at least 5MW or more of electrical output. The input data was also standardized to a -0.8 to 0.8 scale as per the directions contained

Two comparator networks were made :

- A basic modern version which used the Adam optimiser, together with a 'ReLU' activation function
- A "Kitchen Sink" modern version which in addition to the above also had BatchNormalisation, Dropout and Learning Rate decay on plateau applied to it.

As with all the models discussed in this work the network is built using standard Dense Layers as provided by Keras. Keras sets default parameters relatively sensibly incorporate or can take advantage of some of the more useful advances that have occurred in recent years such Glorot initialisation and mixture of 'ReLU' activations for the hidden layers and continuous outputs. In this case initialisation used were changed to 'zeros' which was the more common approach in relation to initialisation for standard ANN's described at the time.

One advantage that was required in order to allow the replication of Fast's architecture to work was that it was necessary to utilise a technique called gradient clipping, whereby the maximum gradient that can be addressed by the gradient descent mechanism is 1. Without this modification it was impossible train the network as the updates appeared to exceeded the capacity of the GPU to handle. It is unclear if this proved any further benefits to the network, however this overall training seemed to be approximate to the results that were achieved by Fast in relation to the 200 epochs being required in order to achieve maximum accuracy.

All models were trained with an early stopping mechanism if there were unable to improve the validation data's data loss.

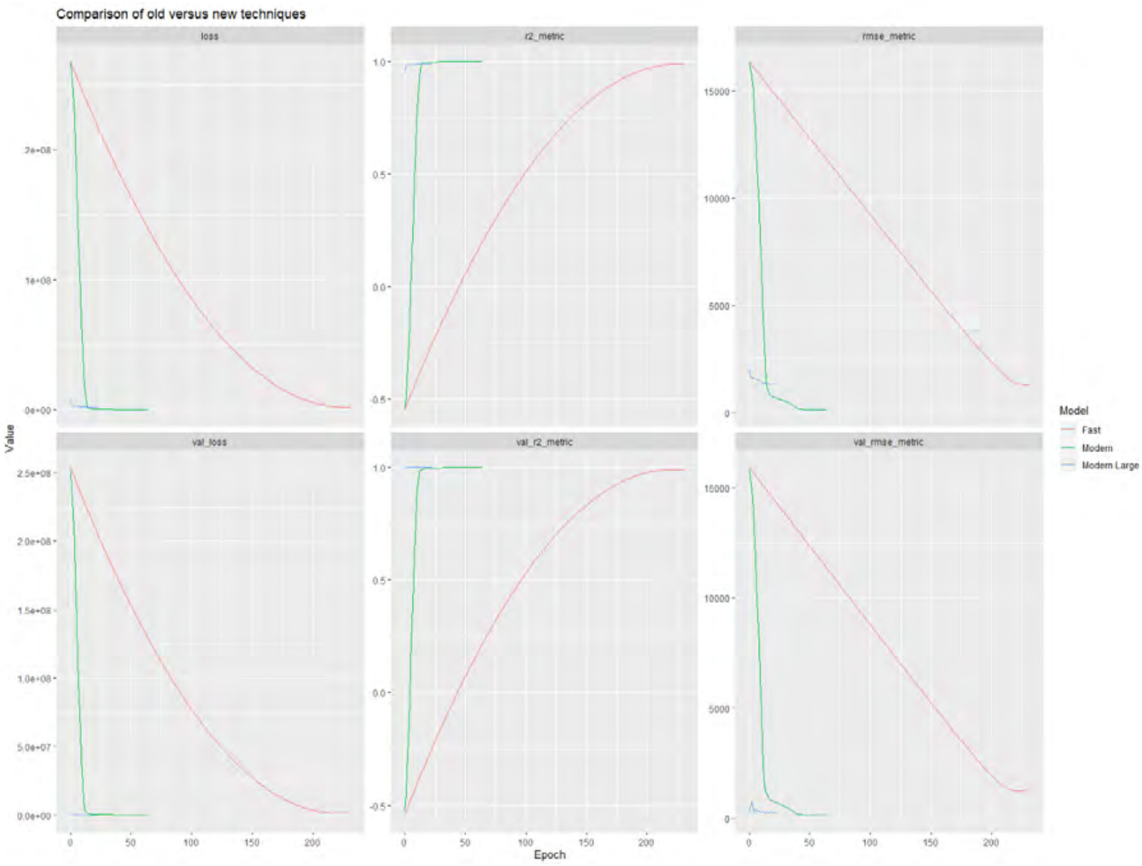

Figure 3.5: Plot detailing the training performance of the various networks in the replication of [Fast et al., 2009b]. As can be seen the modern networks have easily surpassed both in terms of accuracy and speed of convergence (which appears to have also improved from that which was documented

As can be noted by the re-implementation of Fast's original neural network [Fast et al., 2009a] some quite impressive performance gains have been obtained from these advancements in relation to the convergence rates achieved on various networks.

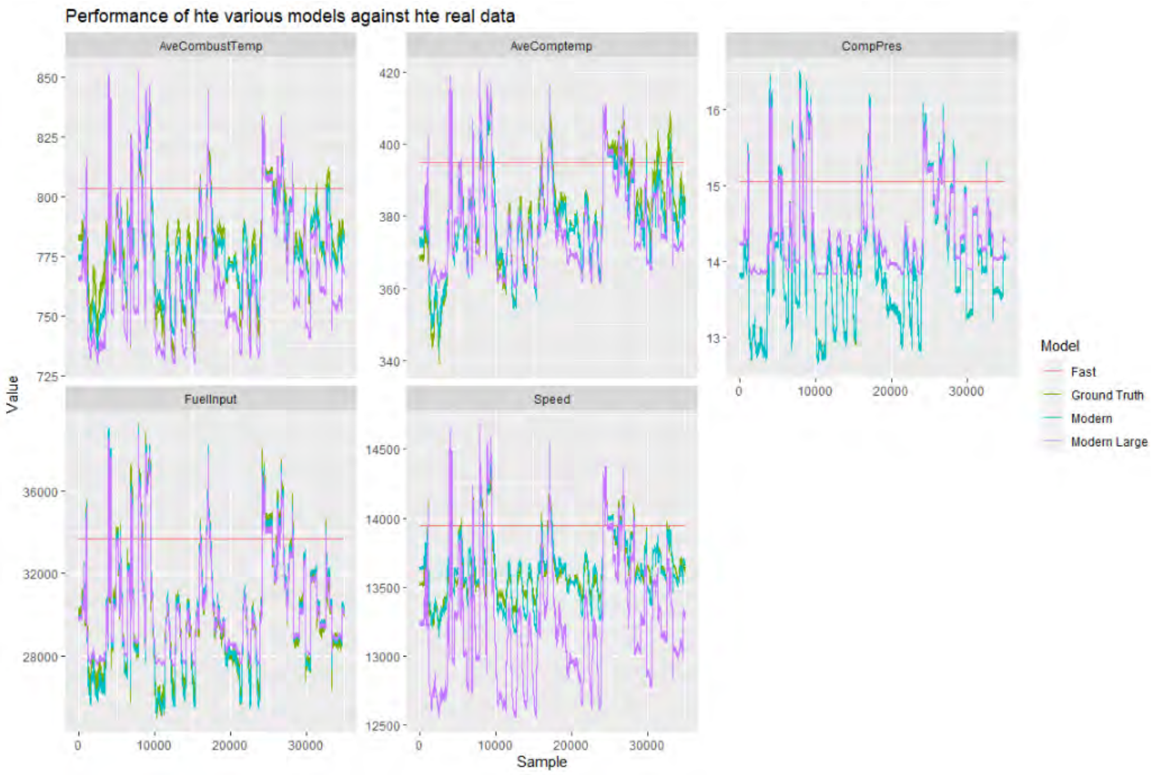

Figure 3.6: Plot detailing the output of the various networks in the replication of [Fast et al., 2009b]

Chapter 4

Overview of previous applications of Neural Networks for IGT

4.1 Introduction

The area of predictive maintenance, despite being a considerable talking point in relation to the marketing material at most of the large OEM's is currently still many years behind the state of the art in relation to the actual application of the technologies. The general theme that has been identified by the author in relation to the work contained within this section is that it seems to be largely limited (with a few notable exceptions) to relatively small datasets, consisting of only a few weeks of running data or limited number of variables. This opens up the possibility of many great advances going forward using more data and algorithms and techniques that can explore that more consistently, A prime example of this was in relation to the [Jia et al., 2015] where a dataset consisting of 10 sets of data each with 2400 across 200 points was considered massive. In real world enginering applications such datasets are relatively small and unrepresentative of the state of the equipment running under real world conditions.

There also appears to be a gap in relation to address one of the key problems seen by operators of IGT's (and indeed any other complex item of industrial equipment) regarding the ability of using the increasing amount of data to better predict overall system availability.

4.2 Work undertaken on similar engine types

In this section we will overview the extent of the work that has been published against the target group of engines (IGT's between 5MW & 15MW and their straightly larger siblings (IGT's between 18MW and 60MW). These range from the application of simple Linear Regression Models Batayev and Onbayev [2018] to the application of Multi-Layered Perceptron Fast et al. [2009a] and more recently techniques such as the use of Autoencoders by [Martinez-Garcia et al., 2019] to identify possible faults within the hot gas path of the engine . A similar approach was used by [Liu et al., 2018] in applying a CNN to the same problem of classification of combustion faults and that has proved successful, but may have difficulty in scaling in the real world due to the variance of combustion systems in the real world (as discussed in 9.9 .

The common themes present in all the above research however appears to be in relation to the modelling, detection and prognosis of faults that are present within the hot gas path of the turbine (As described in all the papers reviewed relating specifically to IGT's. Whilst this is a significant and costly failure mode it represents a relatively small number of the cases which ultimately causes issues for customers regarding directly attributable unscheduled downtime.

An interesting approach was taken by Zhang et al.[Zhang et al., 2017] whereby they combined hierarchical clustering with a Self Organising Map Neural Network (SOMNN) in order to understand the deeper relationships that can occur between different component along the same path. In this case the focus was on combustion and vibration which are both systems where problems in one location will likely have an effect either up or downstream of the process (i.e. vibrations transmitted along a rotor shaft or gas path anomalies). A similar approach although using a generative source to augment the available data was seen in [Macmann et al., 2016].

4.3 Digital Twin

There has been a number of articles that have identified a successful method of identifying and later classifying defects is by way using a model. There are a number of these described within the literature. From the writers personal experience of working in or with remote diagnostics centres of a large OEM manufacture that has a multitude of differing product ranges, the most common model currently is through the use of physics based engineering models of the IGT's. These have often been built up over many years using and utilise well understood methods for the modelling of specific features of the IGT. One of the more recent models described which is being utilised in a live environment is describe by Crux-Manzo et al. [Cruz-Manzo et al., 2018] which describes a live model that self tunes to the turbine that it is connected to, but that was built upon simplified models (albeit no-linear and parametrised) so that they could adequately account for the frequent off-design operation of the majority of gas turbines in real world use.

4.3.1 Signal Reconstruction

An early paper in relation to the modelling of an IGT based on its normal input parameters and then producing a full twin was [Fast et al., 2009b]. A replication using available datasets was undertaken in 3.6

One of the key issues in fault identification and isolation is having a baseline to compare against [Wong and Luo, 2018] used a novel approach by utilising a mass of time delayed recurrent autoencoders in order to replicate a digital twin of every signal that is generated by the engine. Some of the apparent drawbacks of this model is that it results in a very engine (and time) specific model and will require maintenance as it has little understanding of the normal degradation that will occur in normal operation as described in the previous chapter.

4.3.2 Predicting events on Gas Turbines

An interesting approach described by Palau Palau et al. [2018] is using the the Weibull Time To Event (WTTE) architecture first proposed by Martinsson [Martinsson, 2016]. This involves the use of a RNN trained using a survival function for its loss metric. This survival function is calculated by adapting to the population of engines in relation to a known fault condition and the problems involved in relation to dealing with the unbalanced censored data that if prevalent in this domain (i.e the fact that the example are rare and the majority of samples will be will have an end point). An unusual feature of this network is the use of the Weibull distribution (otherwise known as the bathtub distribution). This has a useful property in that it models equipment lifecycles more accurately then many other distributions. Nearly manufactured equipment is an unknown quantity and it is generally perceived that new equipment is more likely to fail when first installed then once it has "bedded in". this is often due to issues such as manufacturing or commissioning defects. Following the bedding in process the equipment will enter into a more predictable state where typical wear and ageing mechanisms take effect. Ultimately however as equipment reaches the end of its operational life it will eventually finally fail.

Palau takes Martinsson's work one stage further and looks towards aggregating the the models of many individual machines trained models so that a fleet model can be produced and generate two way feedback between the two system. This is quite an ambitious goal, and the writer believes that his model described herein represents an alternative halfway house.

The datasets that were used in for this were the PHM08 [Ramasso and Saxens, 2014] and CMAPSS datasets relating to simulated aero gas turbine engines with simulated faults. The intention of these datasets is to allow for the prediction of the remaining useful life of the overall gas turbine. The biggest criticism that can be levelled against them is that they are overly simplistic in relation to the limited number of sensors, operational patterns, frequency of data and fault classes that are available.

4.4 Prediction of Availability

Despite considerable focus on the condition based maintenance elements of IGT performance prediction as shown in the relatively recent summaries produced by [Zaidan et al., 2015; Almasi, 2016; Muhammad et al., 2017; Zhao et al., 2019], there still appears to be a lack of focus in relation to predicting the overall effects upon the customers production rather then then the individual issues.

Due to the issues surround the lack of data available over large periods of times to academics. The author has been involved in supporting previous projects which have touched on this goal previously and although with subtly different objectives Kamarudin [Kamarudin, 2018] reviewed the applicability of a number of traditional machine learning algorithms and statistical approaches based on a dataset of approximately 30 machines with multi-year operating history presented.It was found that there were novel signals that could be used in relation to trying to match the engines health to the anticipated service schedule (such as combustion change).

The typical approach generally is not to look at overall availability but to predict the remaining useful life of a subsystem or collection of subsystems. There have been extensive number of papers and works undertaken on this point, typically with artificial or simulated data (such as the PHM08 or related CMPASS datasets which were also investigated in the research covered in the previous section). These are not entirely analogous to the approach the problem that is being tackled in this work. First of all this work is trying to remove itself from the classification or trajectory of individual failure modes and look at the holistic system and its performance.

One exception to this however is the description of an OEM's use of a tool known as ORAP (operational Reliability Analysis Programm) [Della Villa Jr and Koeneke, 2010]. This is undertaken by capturing and transforming the continuous data generated from the IGT's directly into the system and produce the ISO3977 standards. This however is a rule based system with no inherent predictive capability. It is also one of the few papers that touches upon more then a handful of units from which to collect its data.

4.5 Summary

In summary then, the world of IGT's does not appear to be trailing a blaze currently within the world of the use of deep-learning. There are however some promising developments, and the general theme appears to be that there is relatively little in the way of general purpose datasets that are available for the analysis ans review of academia (the majority of papers dealing with the results of work that has encompassed a single engine or co-located at a single site).

Chapter 5

General Availability Prediction System (GAPS)

5.1 Overview

The General Availability Prediction System is the proposed solution to the questions problems described in chapter 1 as pertaining to the problems faced by operators in predicting the future availability.

Whilst not state of the art in terms of the application of modern deep learning theory and practice it does represent a significant step regarding such techniques as applied to the domain of Gas Turbines and attacks a problem not presently described within the literature (as evidenced by the preceding chapters).

The system (as can be seen in the overview diagram 5.1 receives inputs from a large number of disparate data sources but at its core is composed of three key components that are necessary in order to predict the ongoing availability of the units.

- Configuration Model(s), which identify the likely installed sensors available within the package, and also its configuration.
- General Operative Model, which understands the broader messages logs in relation to their meaning identified by the log embedding and comparisons with the Sensor Driven Operative Model.
- Future Availability Prediction, which is generated from understanding learning the effects on availability of the various extent of deviation from norm that can be observed when reading the current operating data from the previous two models.

5.2 Software Environment

The software used for this included the use of the R [Team, 2013] programming language for the core data gathering, preprocessing and analysis. The modelling of the various deep learning components were build upon Tensorflow[Martín Abadi et al., 2015] with the Keras library [Chollet and others, 2015], initially driven through the Keras R interfaceAllaire [2017], however final models were produced in the native Keras module within the Tensorflow library itself and written in Python. Both RStudio and Spyder were utilised (along with their native notebook solutions)

5.3 Available Data Sources

Every gas turbine collects data from various sensors that are located strategically around the turbine such as important temperatures and pressures. Unfortunately the

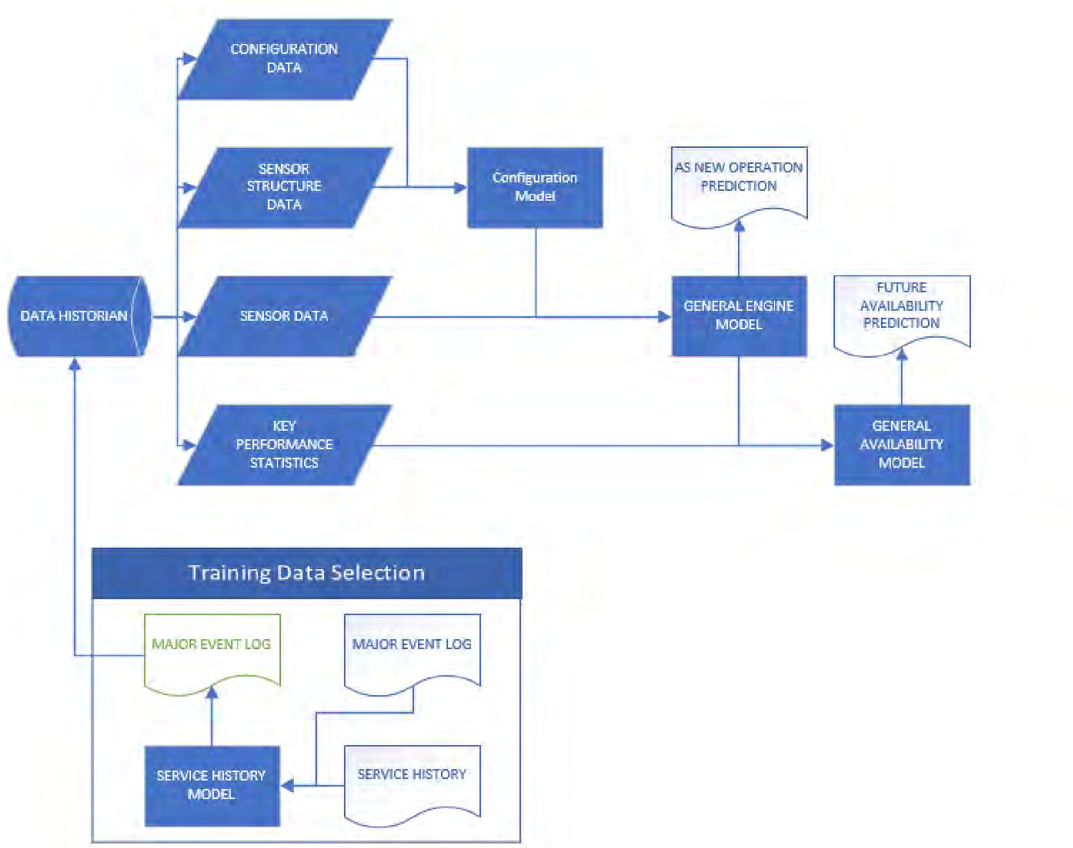

Figure 5.1: Simplified overview of proposed GAPS system

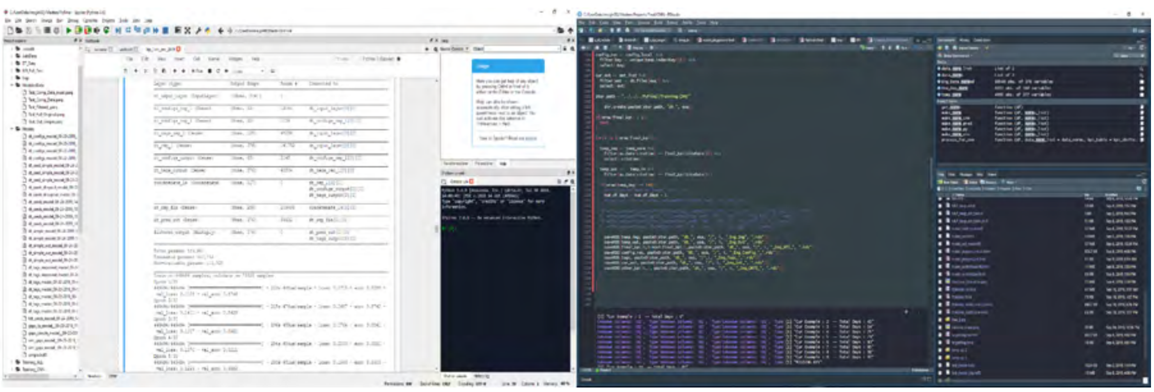

Figure 5.2: Screenshots of the two IDE's used (Sypder and RStudio respectively

majority of these sensors were added purely for being essential to operations or safety and not typically intended for the benefit of condition based maintenance.

There has however been much focus on extracting the maximum value of the sensors present and numerous fault modes can be identified using a variety of techniques which have been deployed.

5.3.1 Sensor Data

Log files are collected on a regular basis from the IGT's Human Computer Interface (HCI) this will be typically at a frequency of once a day or in some case hourly or pushed instantly upon occurrence of unexpected event or outage occurring for immediate attention by engineers.

These sensor logs typically consist of between 80 to 400 different "Tags" or recorded elements which provide discrete reading from sensors located around both the gas turbine (such as vibration and bearing sensors located within the gas turbine) and its supporting ancillaries (such as pumps and gearboxes). the name Tag comes from the physical tags (similar to dogtags) that were once attached to the individual devices with identification information and the terminology has stuck in some quarters. This data is typically recorded at a resolution of 1 sample per minute, with burst of samples at 1 second intervals during key events such as starting up or unexpected shut down. There is not always a one to one relationships between Tag's available and the devices fitted. Some devices will report multiple values so as recieved demand, position or ongoing current draw.

5.3.2 Sensor Structure Data

For the purposes of this model a total of 376 tags have been attempted to be collected from each gas turbine available. Many of these are not available on every unit, or may represent subtly different values (i.e averages over different periods of time), which are difficult to clean up for the 500 individual units contains within the historical archives without potentially affecting other services that depend on that data and have already taken into account any data-integrity / quality issues that may exist. Any Tag where there was a total of 15 or more instances available were included, This reduced the number from a potential 3000 items down to the aforementioned 376.

It should be noted that the majority of Tags that monitor the conditions within the gas path, and thus are used for analysis of the degradation issues identified in 1.3.

In order to help assist the model in relation to this the tag structure available for each unit is one-hot-encoded .

5.3.3 Event Logs

The event logs are produced by the turbines control system to highlight specific issues of concern to operators. These could include items (such as high temperatures or levels of vibration) which would be observable within the sensor data, however they are also adept at highlight items which do not typically appear in sensor data such as boiler failure where typically the control system itself would have no visibility of the state of the boiler other then a single status value sent to it for safety reasons (i.e if the boiler is running dry then it is necessary to shut down the gas turbine as the main external source of heat to prevent damage to a very capital expensive piece of equipment).

The event logs are recorded in real time as the system receives messages and status from the relevant systems so do not always line up obviously with the equivalent sensor data. In the event of a serious event on some systems you might find the spike that has triggered the event message has been missed by the sensor data capture algorithm as it occurred inside the typical 1 minute interval between recording the sensor data. This is another reason why message logs are a vital element in predicting future availability.

The analysis of the event logs is more akin to Natural Language Processing, because although they do occur to varying levels or correlation to events that should be identifiable in the sensor data, they are subject to different formats, hidden set-point (i.e the levels at which they trigger, either high low, or based on something more sophisticated such as a rate of change) an on occasion language (both Spanish and Russian equivalents exist within the dataset).

This has lead to quite a spare training set in some cases, and although the embedding layer that was pre-trained and tokenized upon the identification of the 8700 potential unique tokens within the dataset, the final engines that would be used for training the messages layers were more tightly restricted for expediency for the purpose of this masters.

5.3.4 Configuration Data

This is a high-level overview of the known configuration parameters of the IGT and its supporting ancillaries. This includes details of the engine rating (i.e how many mega watts of power could be output from the unit at ideal operating conditions). This has been hand collated as many of the engines that are contained within the database pre-dated many of the more modern digital configuration tools that are currently employed and are only brought across to modern systems upon the commissioning of a major retrofit to more modern standards.

5.3.5 Key Performance Statistics

The KPI's of a gas turbine are generally governed by the rules set out in ISO 3977-9:1999 and latterly ISO 19589-13:2016. These define the various criteria and nomenclature for describing the various KPI's which this system will ultimately be targeting. Which primarily will be the Available Operating Hours.

5.3.6 Major Events Log

This is the list of major events that have occurred over the past 7 years. It identifies every time that a customer has unexpected requested some additional support (be it parts, labour or both) in order to rectify a situation which has occurred. This log is the record of actions that were taken in order to expedite and return the customer to focus, and as such there is limited data that we can take in terms of diagnosis of actual events (the logs can contain large free text descriptions of events and procedural status by people of a large variety of technical knowledge and communicative ability). IT does however provide a good marker in terms of the seriousness of the events that have been identified within the KPI log described above and allows us to focus on those periods of being of significant criticality.

5.3.7 Service History

The service history for the IGT is a very high level overview of the number of hours typically operated by a gas turbine over its period. Due to the fact that a cumulative digital return of the hours run by a gas turbine is not available and would have to be inferred from potentially very sporadic returns of data (where customer only supply data when under known fault conditions for specific investigations) it was deemed best if an external counter of the operational time was used so as to be more generically applicable.

This was then broken into two groups. One was to identify the overall service period that the engine was anticipated to be within (i.e one of the 6, 8000hrs periods that would typical indicate and engine anticipated service intervals) and a more fine grained counter for the number of hours operated on the day in questions.

These were to be labelled using basic linear models. However ultimately this was not possible within the timeframe of the programme and has been moved to future work. It is hoped that such additional information will greatly enhance the ability of the system to differentiate between different stages of a IGT's life.

5.4 Methodology

5.4.1 Data Extraction

Digital Twin

For the training of the digital twin an investigation was undertaken to identify the most common Tags across the various engines present within the database. This lead to a common core of 370 Tags which are likely to be installed on the majority of units (some tags such as rotational speed had unsurprisingly 100% coverage). Targets for this data were then identified by reviewing the service history to identify newly installed engines as these were considered to be in the closet to new condition. Sample of upto 8 weeks of running or 1500 hours. This was intended to provide a reasonable base line in relation to real operational conditions identified across the fleet.

All available units were targeted with no direct supervision or health assessment undertaken of the individual units. However a check was introduced to ensure that the engines had not been replaced within 6 months which might indicate an early failure or significant fault non operational related fault (the period was deliberately short as most other failure post this period are attributable to operational rather then manufacturing faults, i.e the use of poor or contaminated fuels fuels, foreign object damage or site specific conditions.

A total of 599 periods were identified where some potential data would be available. 450 were selected for sampling within the training set.

GAPS

A slightly different approach was taken in order to identify and extract suitable data for the overall system. A smaller collection of engines was sampled (a total of 8 of the same overall configuration of engine) but various points in their life cycle were identified to be targeted with an even split of "good" (defined as being a period where the engine was available for service for 30 consecutive days, and was expected to be continuing for another 30 days uninterrupted) and "bad" periods (where an engine was leading to a period of downtime within the next 30 days)

Upon analysis if the engine had not been running due to customers instruction it was excluded from the dataset. This lead to a total of 1683 days of training data and appropriate labelling being available.

5.5 Feature Engineering

It has been anticipated that the system would attempt to stay away from general feature engineering in the input data so as to minimise the amount of effort in order to utilise the approach. As such the only transform that has taken place on the data at this time is the reformatting into the regularly sized chunks that have been timesampled down to snapshots taken every ten minutes (albeit it with a variety of measures for the down sampled period being provided as described in later chapters).

This approach (and the use of the CNN) was inspired by the reports in [Khan and Tairi, 2018] of a number of successes regarding the use of CNN's to assist in the generalisation with both noisy data and Remaining Useful Life, which is analogous to the problem at hand.

Chapter 6

Data Driven Digital Twin

6.1 Introduction

As key component of the proposed architecture is the use of a "Digital Twin" of the gas turbine that trained to resemble the condition of the machine as it operates in an near "ideal mode of operation. This approach appears to be common in specific cases where engineering models are available for specific subsystems, but does not appear to have been attempted as a generalised model and utilised for future prediction of availability. Part of the reason is that as noted [Pinelli et al., 2012] there is a disparity between the measurement points used in the best theoretical models and test facilities to those that are typically deployed in the field and reported back through the various remote diagnostic systems available.

6.2 Generalised Digital Twin

One of the elements of the proposed architecture if the "reference" turbine from which the performance of the new engine will be assessed. the architecture of this element has evolved greatly from the initial implementations that were present in the early versions where there was an emphasis on providing as much explicit information to the network as possible.

6.2.1 Initial Architecture

The overall architecture of the digital twin was seen to be relatively simple so as to minimise memory requirements due to the limitations of hardware. Given the general performance of the previous modelling attempt to represent the major non-linearities within the operation of a gas turbine it was felt by the author that Autoencoder (such as that used in Martinez-Garcia et al. [2019] type stack(i.e. one that has in this case a small bottleneck to prevent direct pass through of the data) would be sufficient in order to produce a representative approximation of the idealised version of the IGT. This was was also injected with the predictions for the high-level configuration characteristics and installed equipment structures (I.e. the "Tags")that were actually present for the sensor point. This approach was inspired by the supervised training used in "Greedy algorithms" approach [Goodfellow et al., 2016, chap 8]. This has been used by a number of previous networks to train and optimise layers of a network of a smaller problem and enforce later stages of the network to make use of the previous information in the way intended.

Further to eliminate erroneous predictions from sensors which are not present on the target machine the output layer was multiplied by the predicted Tags model's output which acted as a filter preventing the pass through of such prediction which typically would contain a small element of noise, but would reduce the similarity with the target engines.

This extra information would allow the model to better match the range of possible equipment that it would ultimately be "Twining". This was tested by comparing two equivalent models for test accuracy. The intention of this model is that the weights are to be frozen so that they given a stable version of the relationships between the various parameters measured on the Gas Turbine.

6.2.2 Training Data Selection

The digital twin is built up data taken from newly installed (and thus hopefully in the best of condition). It is trained on samples from 450 such newly installed engines across the spectrum of available engines. The later parts of the systems can then employ other features in order to try and identify the most appropriate failure trajectory that the system is heading down (whether that be through natural degradation or through some more ominous failure mode such as erosion etc).

The engines were selected by filtering the known return KPI data and matching it to known installations where the engine had run more then 8000 hours. This was to ensure that the engine had not had a significant fault from installation. A total of 599 engine across the 5-15MW range of IGT's were then selected and data was extracted. A number of engines had insufficient data out of the target population of 599 leaving 525 with usable data. The timespan of engines used consisted of engines fitted between 2005 to 2018 and could be located anywhere within the globe and

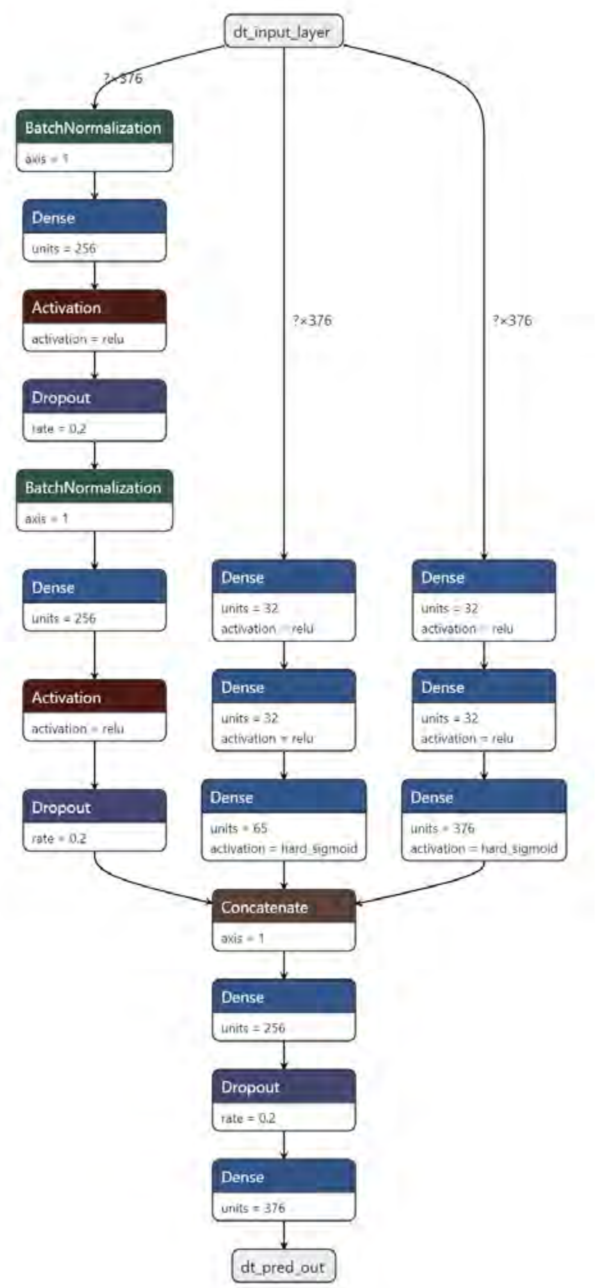

Figure 6.1: High Level of Digital Twin Network

spanning a wide variety of climatic conditions. No attempt to balance the dataset for climatic conditions was made at this stage, but could present future possible improvements.

The data available in the training and validation sets represents approximately 2% of the anticipated running time of the engines (¡ 800 hours out of a potential 48,000 that a gas turbine is expected to endure). It is then sub-sampled across the fleet. This should provide a reasonable estimate (as far as memory permits) as to the operational capability of engines across a broad spectrum of time zones and other environmental climatic conditions.

A total 449494 sample points were used for the training data with a further 79,323 samples taken for the validation data set. This dataset was formed with a 1.35% sample of 450 engines initial running data obtained during the first 30 days of operation following installation (or potentially approximately 600hrs)This was sampled randomly throughout the available data sets. The total dataset amounts to 3GB

The data covers engines from across 4 different product families introduced to market in 1981 and with a wide mixture of configurations (such as fuel and combustion types and operational mode), control systems (some based on control systems first designed in the Mid 1980's and thus exhibiting different design philosophy for control of the turbine). his traditionally has been seen a major stumbling block for the development of larger models within the industries. Even the current best-of-bread have significant limitations in relation to there ability to monitor issues such as transient changes or off-design operational modes.

It covers a total of 376 potential measurements points that are available (some are essentially the same point but expressed in subtly different ways in different system configurations). This leads to the possibly as utilised in some forms of the twin to infer the some of the potential configuration options that are available. When measurements were not available for a given engine, this value was replaced with a Zero. These measurements cover items such as :

- Temperatures across the gas path of the engine, bearings, and ambient conditions.
- Value positions (both demanded and feedback through independent sensors)
- Flow rates and pressures as various points
- Vibration
- Rotational Speeds

The initial design concept involved the augmentation of the core sensor data with details of the sensors used and the high level configuration details of the engine (such as product family and overall rating). This however proved to be not required as some initial tests showed a high degree of predictability due to the nature of the way the data was extracted, i.e by obtaining a comparable data structure for each engine and then using that to identify if the engine was of a particular configuration (which itself is often based on the anticipated tags available). This allows for the network to much more closely map to the engine. It also aligns more closely with the goals of the model being an End-to-End model which is simply presented with the data and from which it can make direct assumptions. This has allowed for a completely generalised model to be provided that works on the whole family of Siemens SGT100-400 class turbines.

For training purposes a Tag Matrix was produced which would identify by inference from the mean value of the entire dataset available for a given engine whether or not there would not a individual Tag was available. The configuration data for each site was taken from the central configuration database. Features included, Engine Rating, Mode of Operation and Package architecture (which will relate to significantly to the equipment fitted and the anticipate operational role of the engine)

At this stage it is not seen as necessary to greatly increase the amount of data due to the relatively early convergence and generally acceptable performance of the simpler models. There will however be opportunities in the future to greatly increase the machine available for the development of these models by moving the training elements into the cloud. By utilising big data type infrastructure this dataset could be greatly increased as there is potentially 7TB of target machine data available within the existing remote diagnostic infrastructure spanning over 500 engines that have at some point provided information. Unfortunately this infrastructure was not available to be used during the research presented in this thesis.

6.2.3 Configuration and Tag Prediction Models

These sub models was built using the functional API which allows for the separate sub-models to both be trained independently and also allows for the reuse of layers and model easily in a object orientated fashion. they essentially follow the same overall architecture, though they have bee tuned slightly to the varying amount of data that they have to represent.

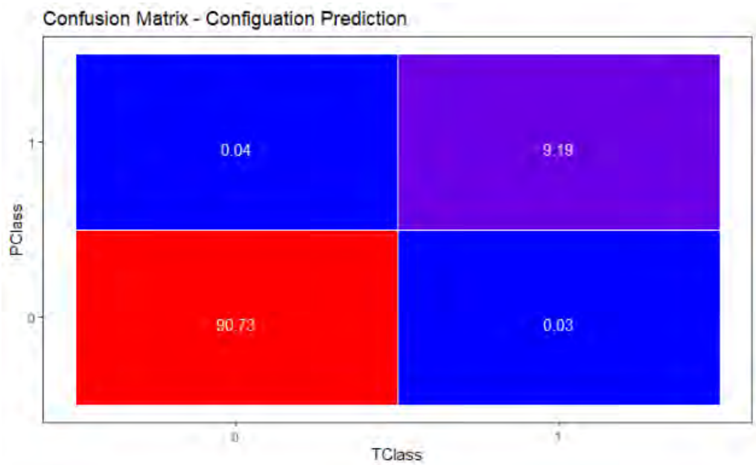

Figure 6.2: Confusion matrix for the classification of configurations parameters of the target engine. all figures quoted as percentages

Configuration Model

The configuration model handles the prediction of the higher level features of the gas turbine such as engine rating (i.e it's MW output) or the main features of its supporting package (such as as general architure for supporting auxilliaries such as

As can be seen from the confusion matrix (6.2 and 6.3) the overall accuracy of these models is very high. As the every sample used has not been directly inspected and verified it is highly likely that a substantial portion of the anomalous entries are found to be errors in the original configuration in the core database (such as tag's configured that are not installed on the transmitting unit and thus which no actual data is being recorded against, but for which a 0 is being inferred through some process within the pre-processing pipeline).

It was somewhat surprising for the author that the system was capable of producing a model that was capable of identifying 99.93

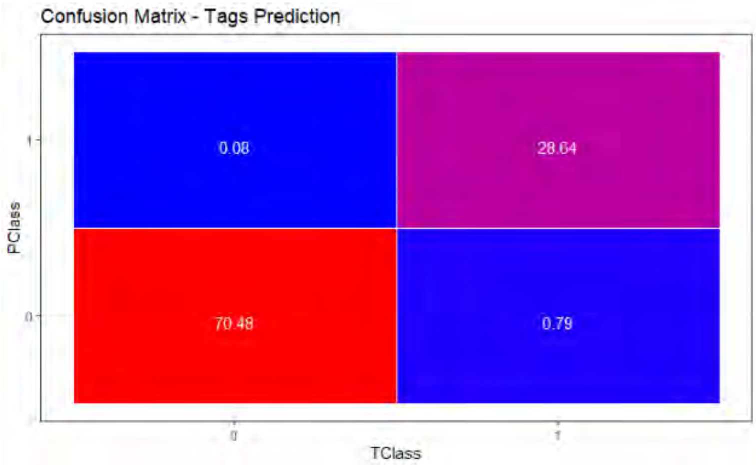

Figure 6.3: Confusion matrix for the classification of Tag parameters of the target engine. all figures quoted in percentage

6.2.4 Message Logs

It was originally anticipated that Message Logs could be also integrated within the current architecture. However due to the significance in-balance in data relating to failures modes that are identified from Messages to enable this project to remain within the scope of a Masters this element was removed. The initial prototype entailed utilising an embedding layer to encode the messages received during a single interval so that they would be available for the model to assess. This did appear in a few cases to be learning (as convergence was observed to some degree on engines with very similar message output), however as described previously the scope of activity covered by messages and the variety of them proved to be unsuitable for work at this point and it would potentially limit the ability to demonstrate a generalised model.

6.3 Illustration of Model Representativeness

During the training of the model the digital twin a version was re-targeted to predict actual values rather then the normalised values. This allowed the output to be compared to that of a true Gas Turbine. As can be seen the results are largely comparable. One combustion can (represented by the dual trends seen displaced on TC8 and TC9) appears to be being significantly misaligned, however it is also noted that the model has not identified the "swirl" caused by running an engine faster, whereby the relationship between the combustion cans and the later thermocouples presented here shifts slightly. To a diagnostic engineer the combustion output would look reasonable. It is also worth noting that the model has correctly ignored the TC14- TC16 which were not fitted to this particular model.

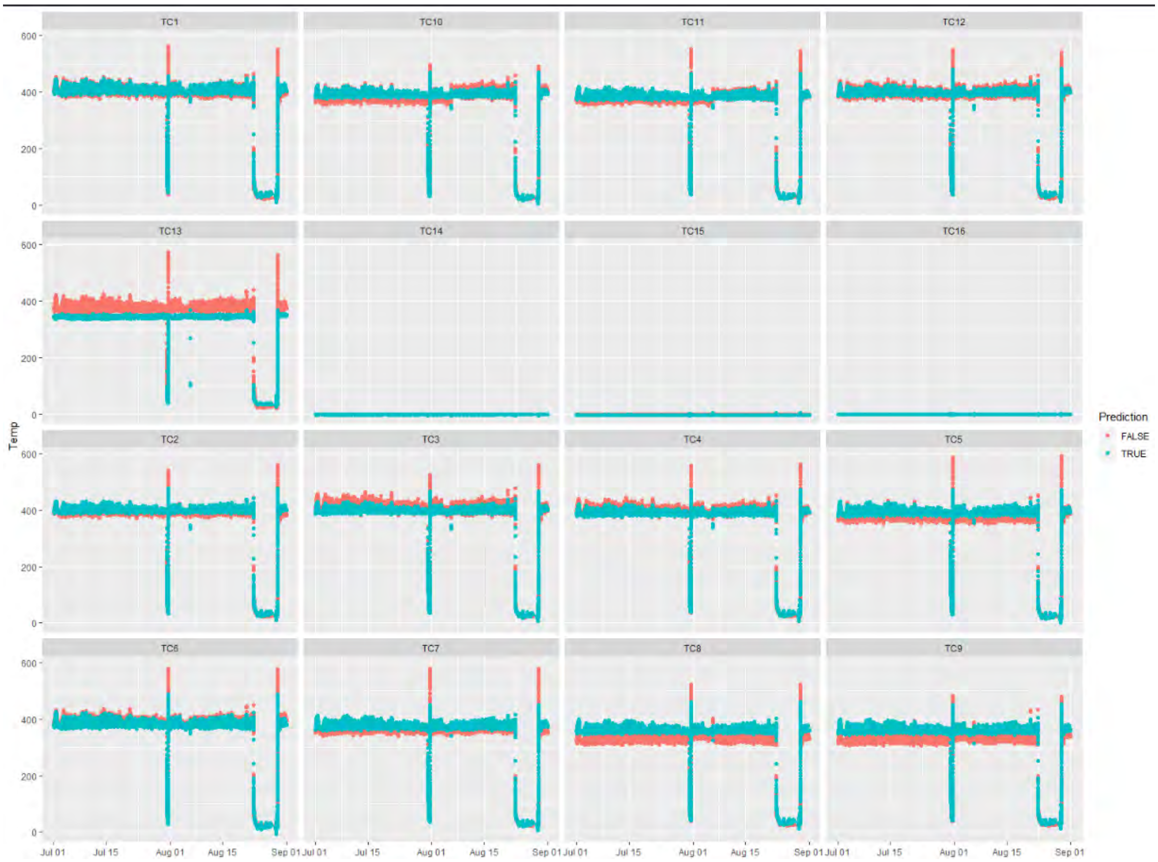

Figure 6.4: Overview of Combustion Output of Model

Chapter 7

GAPS Model

7.1 Overview

The GAPS model is based on previous research into prediction of Remaining Useful Life with the use of CNN's. It is augmented by having the ability to compare with a digital twin in order to identify see the different between an idealised gas turbine and the unit that it is currently reviewing.

There is an issue in relation to the balance of data that is available in this scenario, however that would be overcome in reality with the undertaking of the system to have access to the full on-line database of data.

7.2 Data Preperation

The data for GAPS takes a slightly different approach to that which was used in the digital twin as the GAPS looks at the overall operating profile of the IGT over a reasonable period of time (in this case a day). This allows it to infer the state

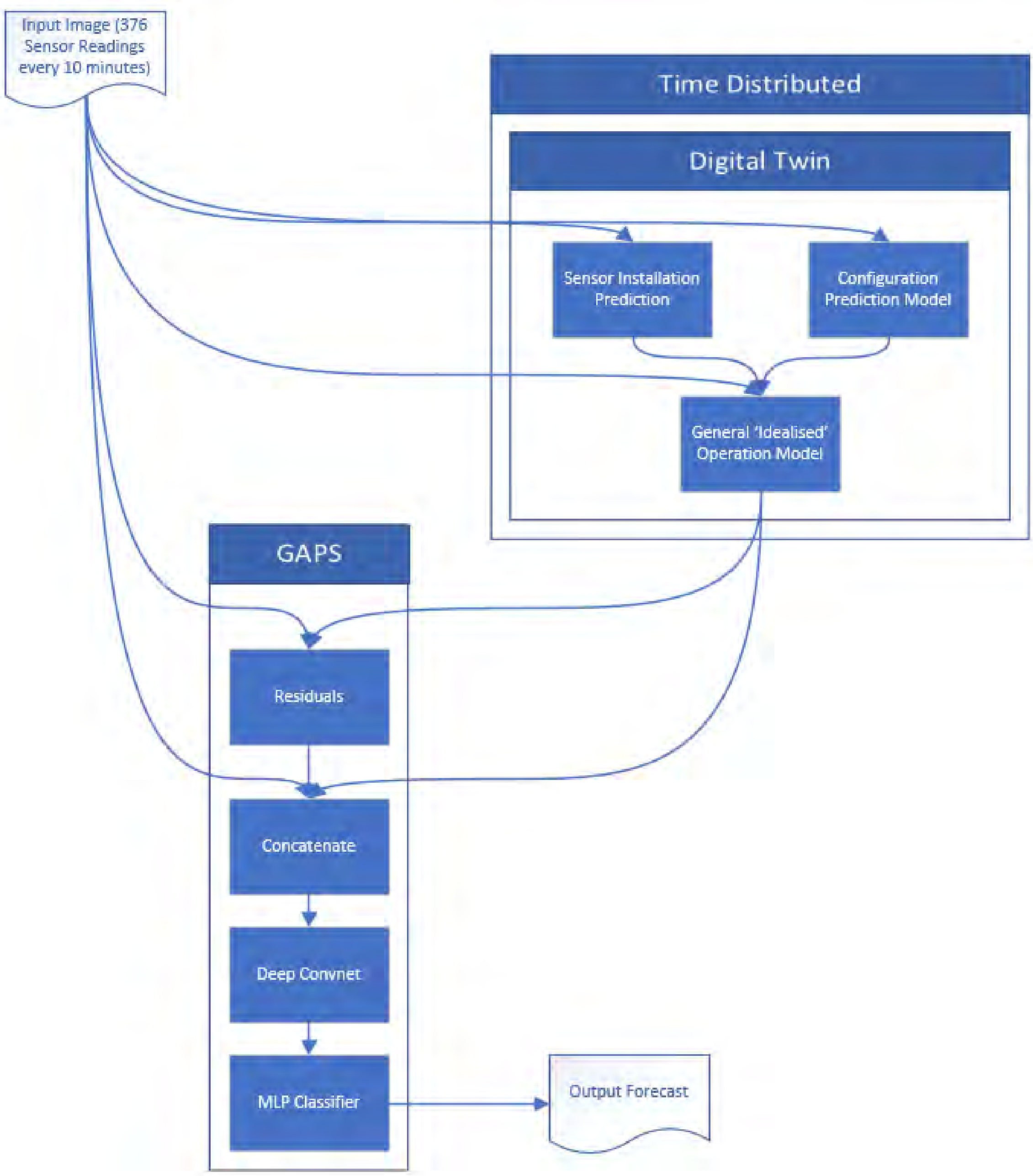

Figure 7.1: High Level Overview of System

of the turbine and a large number of its ancillary components. This view can be visualised as in 7.3 Each vertical line represents an individual sensor or other discrete measurement point available from the engine. The horizontal axis represents that time steps (in this case two days so as to highlight the changes on operations that should be visible to the network). A further plot of the running regime of the engine over that period has also been included 7.2 so as to highlight the operational status changes of the engine during that period, and which clearly line up to the lighter patches seen generally in the upper middle of the graph.

The data presented to GAPS has been pre-formatted into 144 samples of data across 376 potential data-points and re-sampled at 10 minute interval's. This was chosen simply to as a reasonable balance between number of available points (typically 1440) and the computational power available. The data was then normalised in line with the preparation of the data used for the digital twin, with the same scaling s applied.

One additional critical point is that the 144 summary of data actually consists of 5 channels of data representing the mean, max, min, standard deviation and median values that were seen over the 10 minute period. This give substantially more information for the model to utilise in relation to looking for features of interest.

Further options in relation to the representation of the data presented to the network are explored later, however and example of the data as presented to the network is shown in 7.3 with the associated running trend presented in 7.2. You should be able to make out that the light areas of the data approximate to the areas seen on the running trend. This plot effectively shows the relationship between the 376 sensors presented to the network react to changes in state.

A time interval of a day was chosen so as to capture the typical daily operation cycle (which can very greatly in some scenarios such as desert installations where the change between day and night can be up to 30°)

Data required as the majority of trends that are anticipated to be caught will occur over a significant period of time. In comparison to the full extent of data available the sample is minuscule, however the intent is to prove out the concept and hopefully show that meaningful information can be extracted. Should the method prove useful other augmentative or generative methods could be used to increase the pool of data available for training. In reality it would be expected that 1 minute (or better in some cases) data would be available.

Data selection for the GAPS training set was undertaken by firstly splitting the range of engines into there 4 basic product families and then selecting 26 periods apiece represent representing good availability periods (i.e those which have been 100% available) and 26 further periods where there engine has suffered a period of less then 100% availability due to assessed "Forced Outage Hours" (which are those which are incurred due to known faults and not by regular maintenance of site shut-down).

For each of these point 45 days were selected. This makes a total of 9360 days possible days of of examples to train upon. In reality only 8240 were available as engines which were down for planned maintenance or customer driven reasons were excluded (a long with the odd anomaly in relation to KPI data not being available on occasion)

Test data was extracted in a similar format but in this case it was restricted to just two engines but over a long period of time (in this case two years each) This

Figure 7.2: The running profile of the engine show in 7.3

leads to almost 1400 days (thanks to some exclusions for the same reasons as above) of real world testing to evaluate the model against.

7.3 Example of Encoded Data

7.4 Architecture of the GAPS

The structure of GAPS network is presented in 7.4 and follows the overview seen in 7.1. A single image is presented to the network as described above. That is then Time distributed along its time axis to the digital twin where every value is then computed in line with the description in the previous section. The output from the Digital Twin is then un-timedistribed and reshaped to be presentable to the CNN. A residual of the difference between the two images is prepared and then all three "images" (i.e the input, the twin and the residuals) are sent to the CNN portion of the network where feature extraction takes place. This is then flattened after going

Figure 7.3: Example of data presented to network

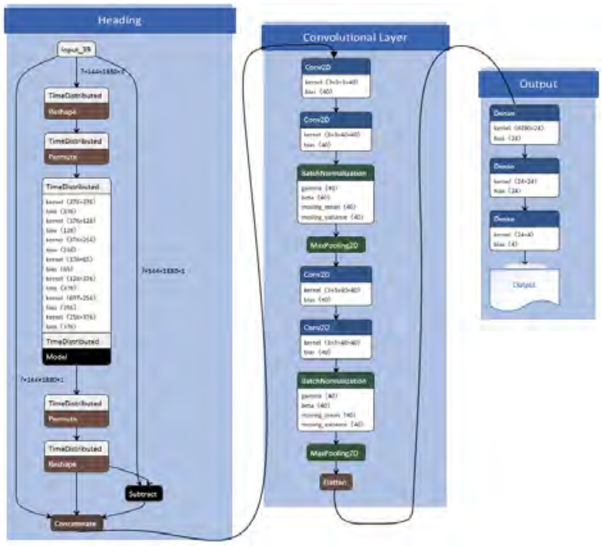

Figure 7.4: Layered Architecture of GAPS system

through eight layers and and a result is computed in the fully connected final layers in order to produce a a binary indication of whether or not the engine is likely to be available over the next 1,2,3 and 4 weeks.

7.5 Training the network

The network is trained by presenting 5 shuffled images to the model per batch. The gradient is then calculated after each batch and the weights updated. This is in part due to the constrains of the hardware that are being used which would have been unable to sustain larger batches (3.5GB of memory available on the GPU). Despite the reasonably moderate number of trainable parameters and the reasonable performance of the training algorithm.

```
Model: "Gaps_Model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_39 (InputLayer)           [(None, 144, 1880, 1 0

time_distributed_44 (TimeDistri (None, 144, 376, 5)  0           input_39[0][0]

time_distributed_45 (TimeDistri (None, 144, 5, 376)  0           time_distributed_44[0][0]

time_distributed_47 (TimeDistri (None, 144, 5, 376)  634849      time_distributed_45[0][0]

time_distributed_48 (TimeDistri (None, 144, 376, 5)  0           time_distributed_47[0][0]

time_distributed_49 (TimeDistri (None, 144, 1880, 1) 0           time_distributed_48[0][0]

subtract_3 (Subtract)           (None, 144, 1880, 1) 0           input_39[0][0]
                                                                 time_distributed_49[0][0]

concatenate_4 (Concatenate)     (None, 144, 1880, 3) 0           input_39[0][0]
                                                                 time_distributed_49[0][0]
                                                                 subtract_3[0][0]

conv2d_17 (Conv2D)              (None, 144, 1880, 40 1120        concatenate_4[0][0]

conv2d_18 (Conv2D)              (None, 144, 1880, 40 14440       conv2d_17[0][0]

batch_normalization_13 (BatchNo (None, 144, 1880, 40 160         conv2d_18[0][0]

max_pooling2d_11 (MaxPooling2D) (None, 36, 470, 40)  0           batch_normalization_13[0][0]

conv2d_19 (Conv2D)              (None, 36, 470, 40)  14440       max_pooling2d_11[0][0]

conv2d_20 (Conv2D)              (None, 36, 470, 40)  14440       conv2d_19[0][0]

batch_normalization_14 (BatchNo (None, 36, 470, 40)  160         conv2d_20[0][0]

max_pooling2d_12 (MaxPooling2D) (None, 4, 58, 40)    0           batch_normalization_14[0][0]

flatten_21 (Flatten)            (None, 9280)         0           max_pooling2d_12[0][0]

dense_43 (Dense)                (None, 24)           222744      flatten_21[0][0]

dense_44 (Dense)                (None, 24)           600         dense_43[0][0]

dense_45 (Dense)                (None, 4)            100         dense_44[0][0]
==================================================================================================
Total params: 903,053
Trainable params: 268,044
Non-trainable params: 635,009
__________________________________________________________________________________________________
```

Figure 7.5: Model Summary for GAPS

Chapter 8

Results

8.1 Introduction

In this section we will review the results that have been generated by the different experiments that have taken place on the way to the construction of the proposed system.

8.2 Tag Prediction

A number of different combinations combinations of structures of network were used as described in the previous section. Two key differences were explored in this particular experiment. The first was again comparing SGD with Adam, but also applying dropout. The interesting surprise here was that the best performing network was in fact the small Adam optimised network. Further over the relatively short number of Epochs upon which training took place it would appear that its generalisation ability was in fact better then that of the other other networks event when utilising

Model Name	Epoch	Training Accuracy	Training Loss	Training R2 Metric	Validation Accuracy	Validation Loss	Validation R2 Metric
LargeAdam	4	0.993	0.105	0.965	0.992	0.119	0.96
LargeSGD	3	0.92	1.123	0.62	0.884	1.701	0.44
SmallAdam	14	0.994	0.067	0.972	0.994	0.066	0.973
SmallAdam Dropout	9	0.944	0.146	0.813	0.969	0.094	0.886
SmallSGD	3	0.829	2.569	0.171	0.831	2.55	0.179

Table 8.1: Overview of final training results. Yellow indicates the winning parameter

Model Name	Epoch	Training Accuracy	Training loss	Training R2 Metric	Validation Accuracy	Validation loss	Validation R2 Metric
LargeElu	19	0.996	0.062	0.952	0.996	0.067	0.948
LargeRelu	4	0.991	0.143	0.888	0.99	0.15	0.883
SmallElu	8	0.999	0.02	0.984	0.999	0.02	0.984
SmallRelu	5	0.995	0.083	0.935	0.994	0.087	0.932

Table 8.2: Overview of final training results. Yellow indicates the winning parameter

droupout).

One explanation for this could be the judicious use of early stopping which may have prevented the networks ability to reach a true global minima, however given the relatively high accuracy these networks appear to be pretty close to idea.

8.3 Configuration Prediction

A simple experiment was also used in relation to the configuration model where large and small networks were trained with either ReLU or elu activation functions. It would appear that in this case the elu activation did in fact produce the best performance. Again as with the Tag predicator above the small network consistently provided better scores than either formulation of ReLU's or the larger elu. This would indicate that both issues (Tag and configuration prediction) should be fairly differentiable from the core dataset.

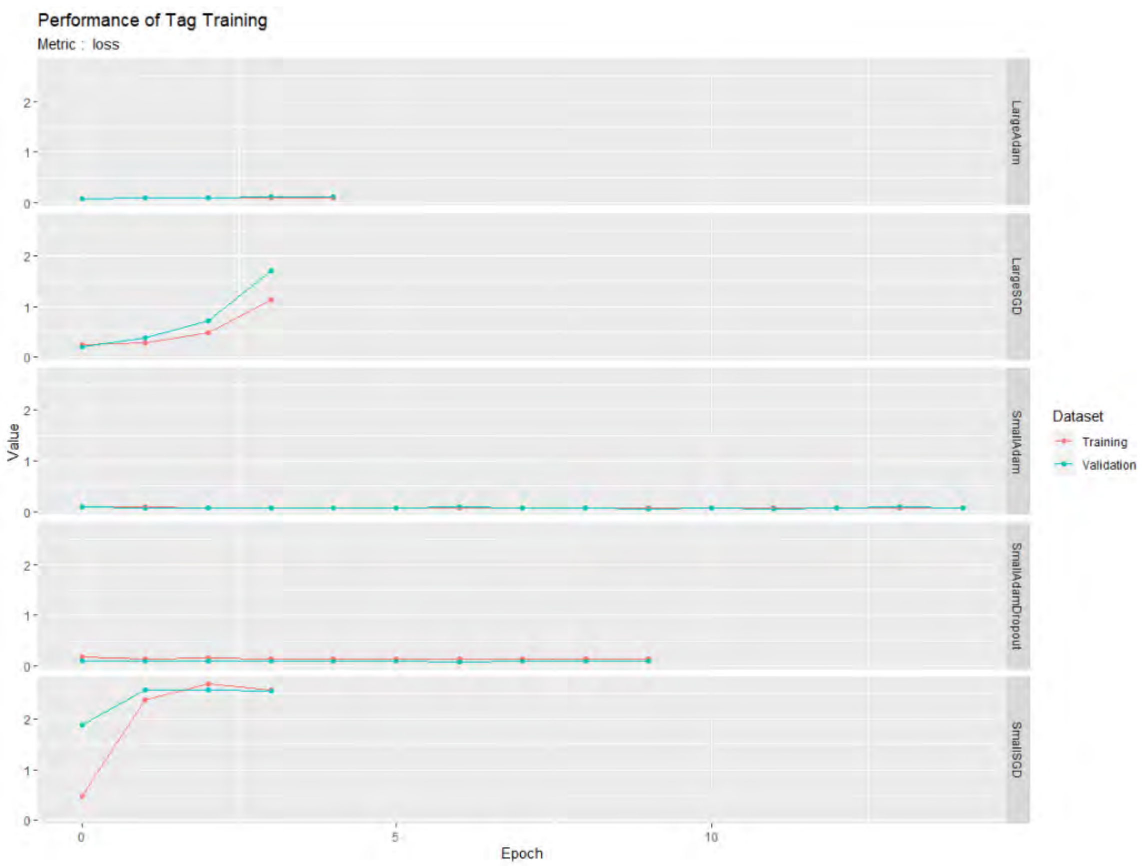

Figure 8.1: Tags : Losses of various network configurations

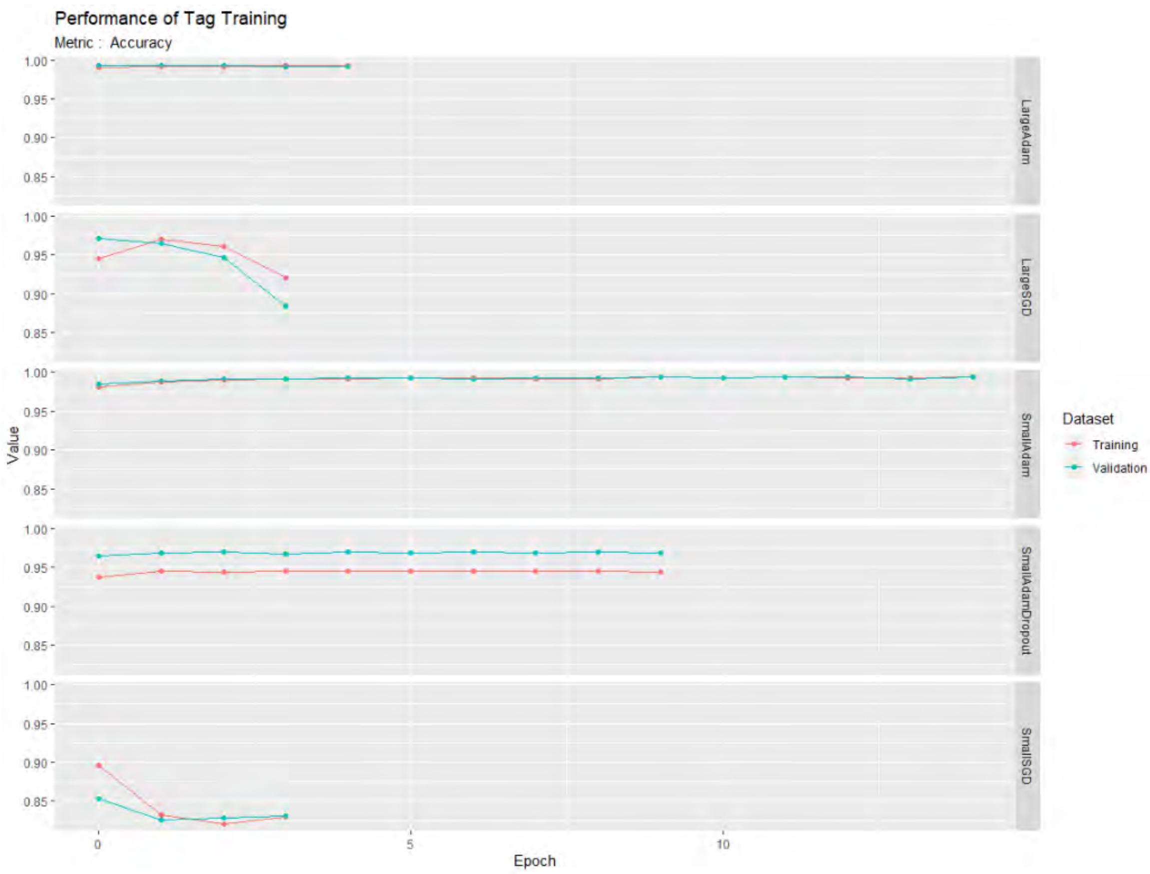

Figure 8.2: Tags : Accuracy of various network configurations

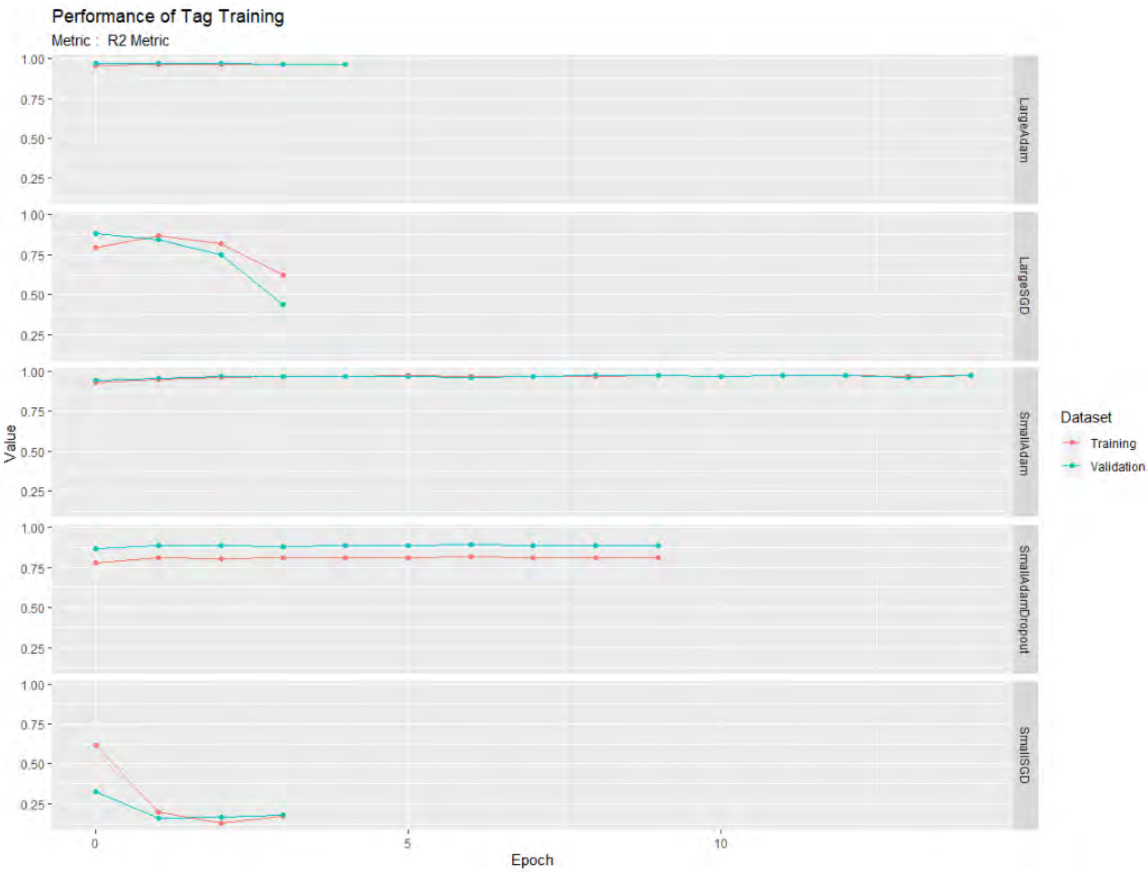

Figure 8.3: Tags : R2 of various network configurations

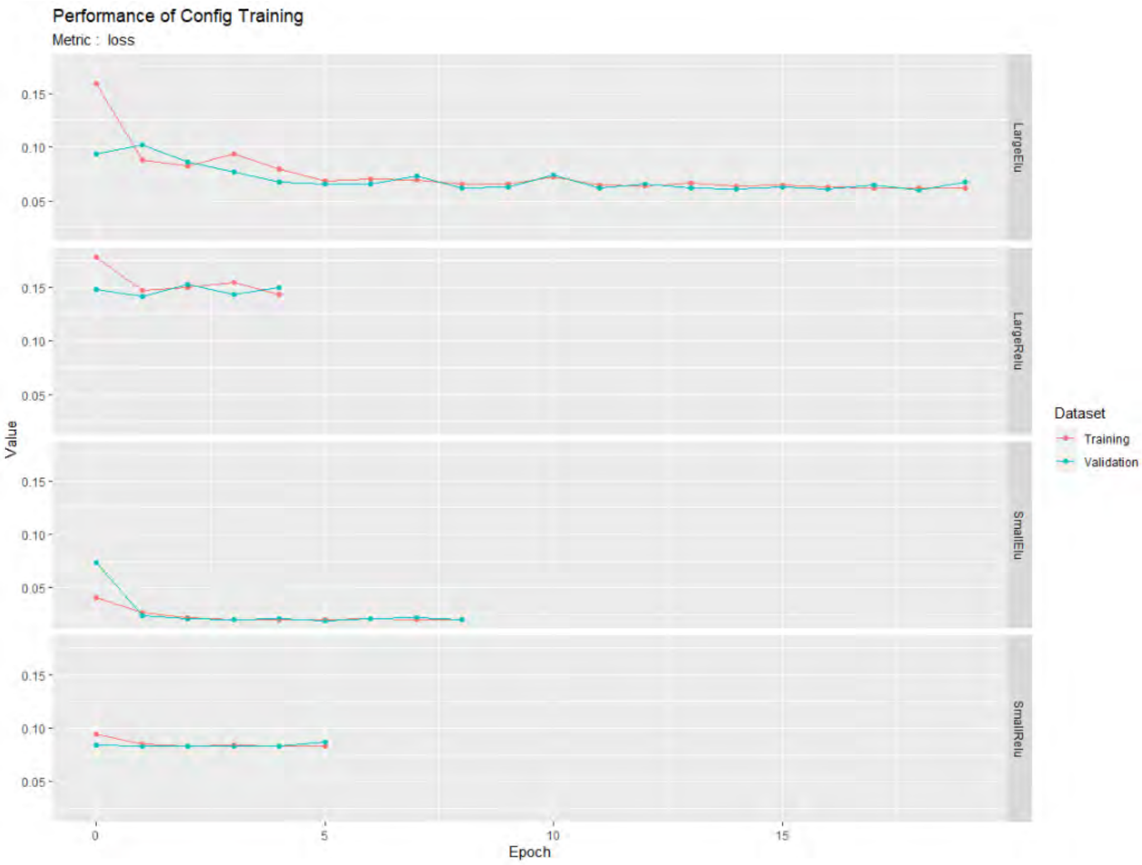

Figure 8.4: Config : Losses of various network configurations

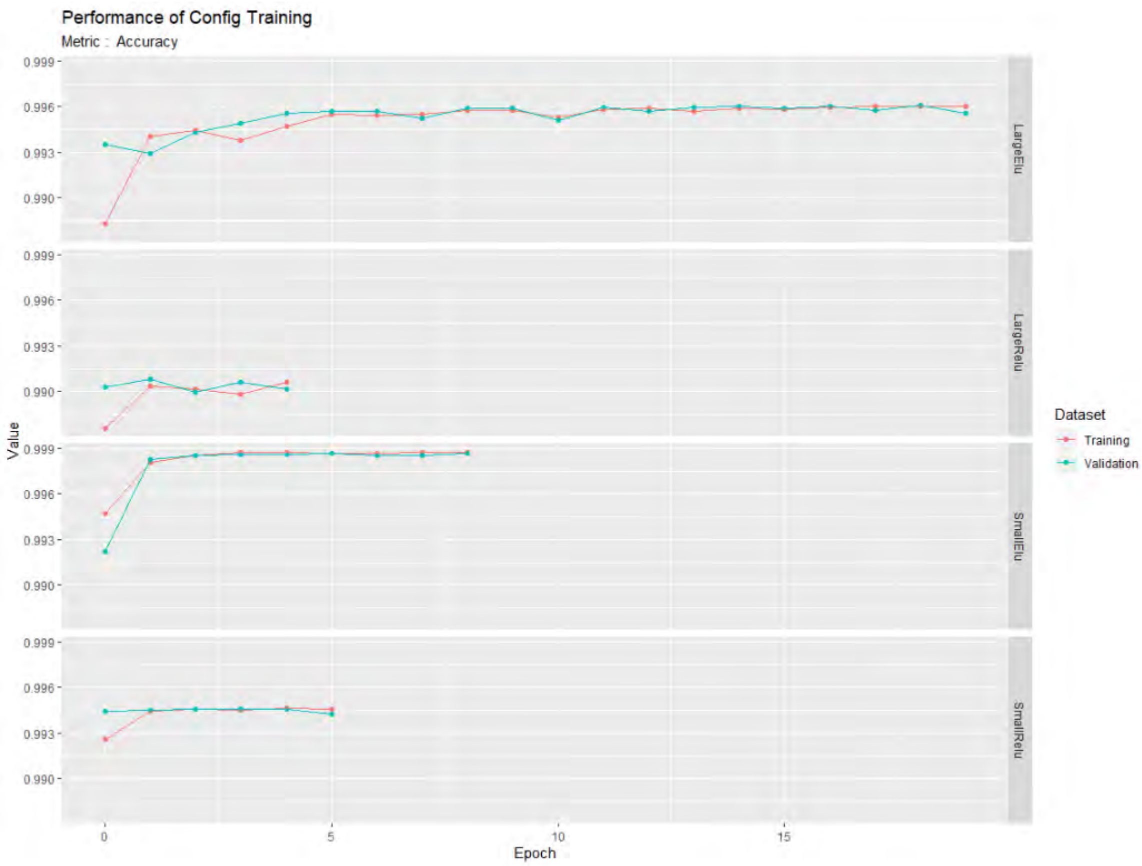

Figure 8.5: Config : Accuracy of various network configurations

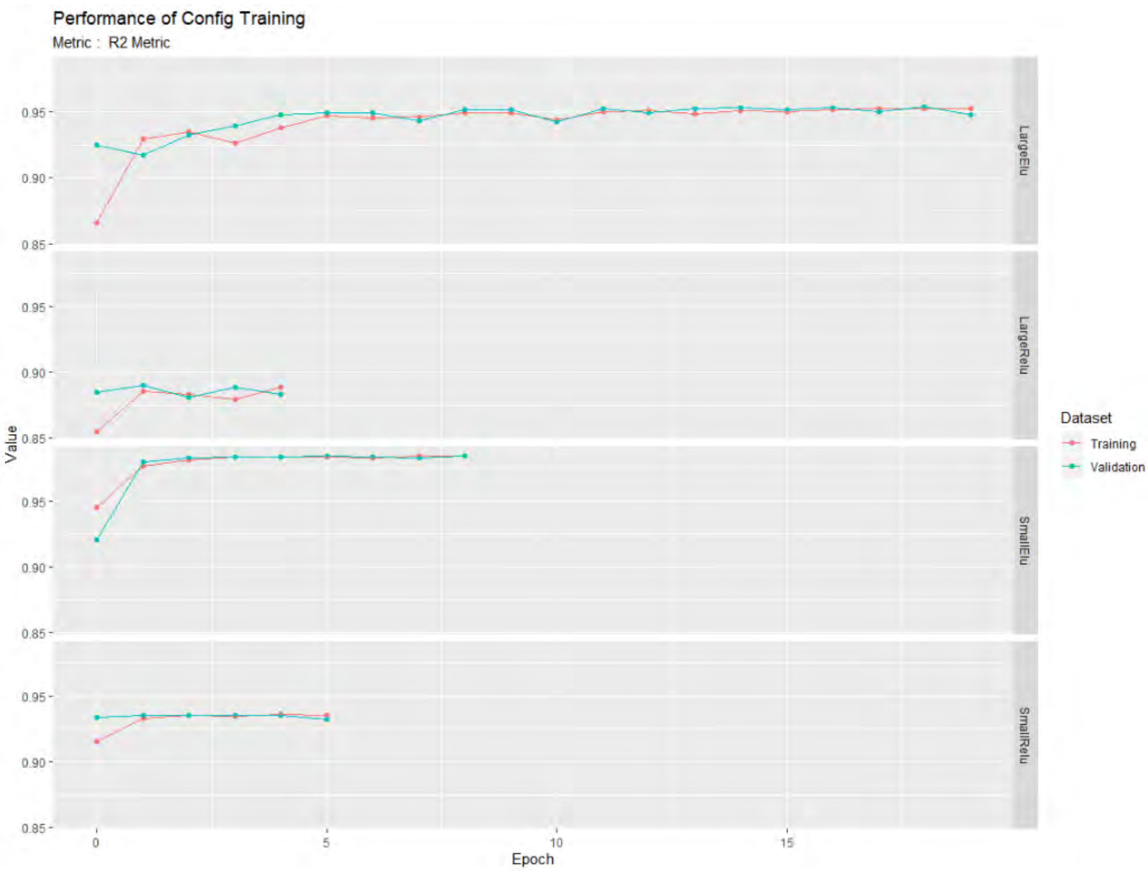

Figure 8.6: Config : Accuracy of various network configurations

Model Name	Epoch	Training loss	Training R2 Metric	Validation loss	Validation R2 Metric
DTM_With_Tag/Config	9	0.153	0.909	0.092	0.91
DTM_Without_Tag/Config	9	0.165	0.895	0.105	0.895

Table 8.3: Overview of final training results. Yellow indicates the winning parameter

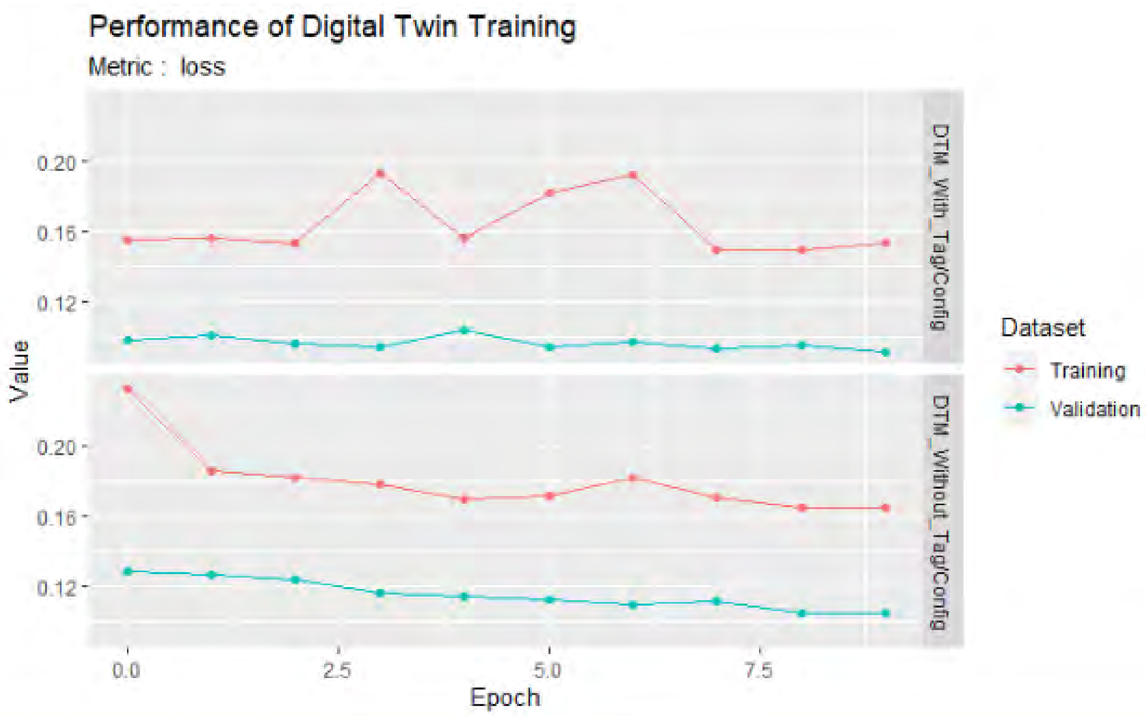

Figure 8.7: Digital Twin : Losses of various network configurations

8.4 Digital Twin Training

Two versions of the digital twin were produced. One with the benefit of the Tag and Configuration information and one version without. As can be seen from 8.3 injecting the Tag information into the twin leads to improvements in convergence time and improvements in relation to accuracy. However it may be that the information can ultimately be extracted with near no-difference in loss is allowed to train to its ultimate conclusion.

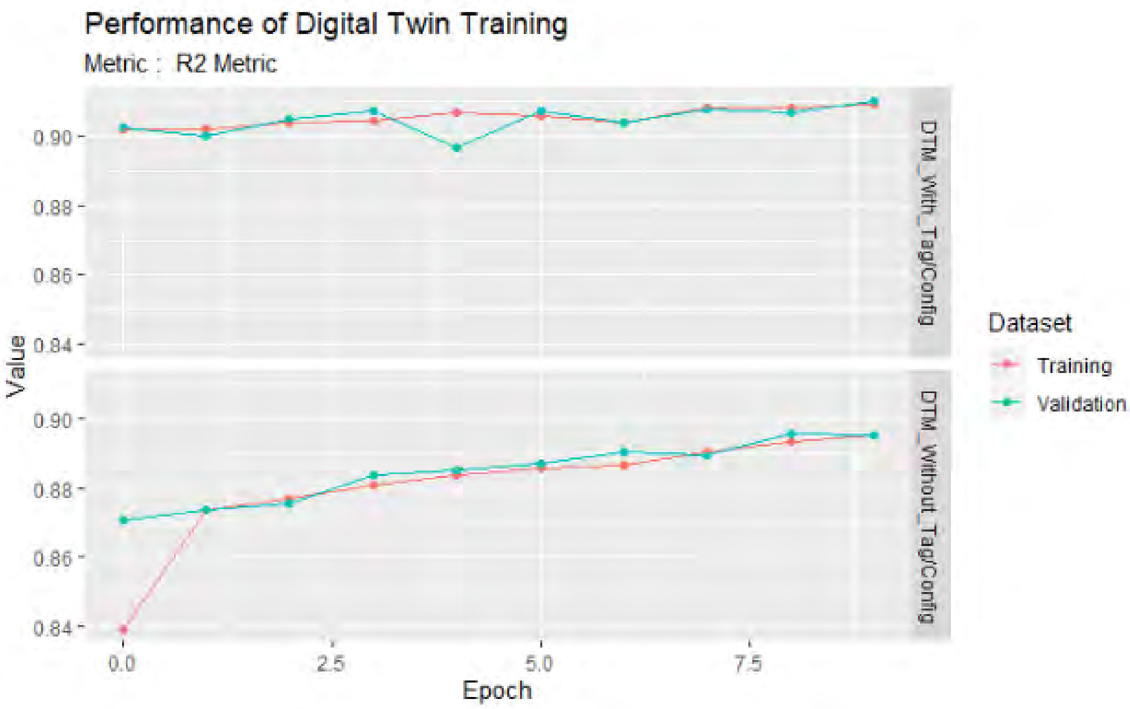

Figure 8.8: Digital Twin : R1 of various network configurations

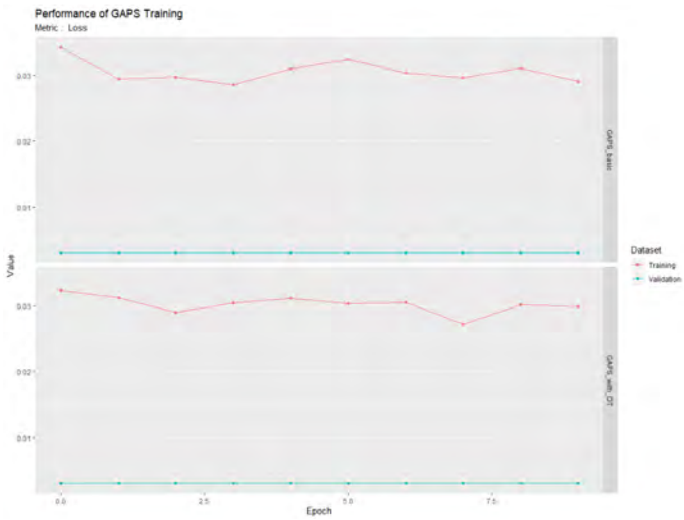

Figure 8.9: GAPS : Training Losses for the GAPS models

8.5 Evaluation of the GAPS Models

It should be noted that both models appears to ultimate begin to capable of learning a representation of the data but ultimately may have needed considerably more time training time then was available.

The baseline model was simply the GAPS model without the input from the digital twin.

It would appear from the data also that the validation data set was not sufficiently mixed with different use cases which partically explains the consistently high accuracy rate in spite of the variability in losses.

8.5.1 Output From GAPS

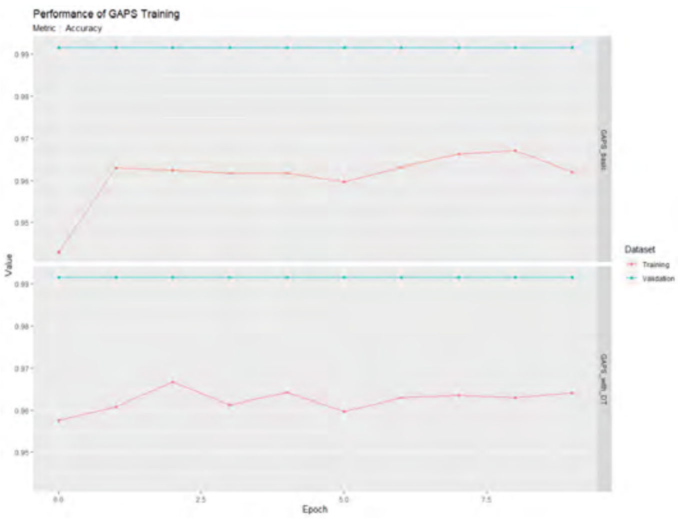

Figure 8.10: GAPS : Training Accuracy for the GAPS models

Figure 8.11: Model Output

Chapter 9

Conclusion & Future Work

9.1 Achievements of Goals

9.1.1 Objective 1 : Build a sufficient data set

This took a considerable amount of time to achieve, but ultimately resulted in a structured and labelled 30GB dataset for the production of the digital twin and 3GB dataset for the predictions of long term availability. These datasets in themselves have significant value in future research.

Objective 2 : Replicate of previous research

In updating the work undertaken by [Fast et al., 2009b] to use more modern techniques and algorithms and showing the substantial effect that they have had in relation to new activation functions, BatchNormalisation and optimisers such as Adam which allow for the vast improvement in the training process for Deep Learning networks. This was also a useful exercise to prove that the work undertaken previously was useful in areas where less restricted data was available and showing testing many of the assumptions that lead to this research were well founded.

Objective 3: Develop a model for the prediction of high level configurations

This was achieved and provided for a high level of accuracy as show

Objective 4: Develop a model for the prediction of available Tags

This was also achieved and provided for a high level of accuracy

Objective 5 : Develop a generalised operational model for a non-specific IGT

The creation of a data driven digital model is achievable, and in a generalised fashion that does not required retraining on individual units, and still gives good real world results.

Objective 6 : Develop an availability prediction model for IGT's

As has been demonstrated previously, whilst a working model has not yet been achieved, it has been shown that there is a information available in the dataset from which a network of a more suitable architecture could potentially be produced from.

Objective 7 : Identify the limits of a generalised prediction system

It has been demonstrated that the limits in relation to any such prediction system are predicated on the fact that only trends that are visible through the known sensors or the behaviour of the engine could be predicted (even indirectly such as repair times for problems). This is no however going to be able to suitable outside influences such as fuel interruptions or foreign object damage.

Objective 8 : Identify future enhancements

as can be seen in below there are plenty of avenues which will allow for further development of this topic in the future. Not only through methodology improvements, application of cutting edge techniques but also the addition of future data sources (and quantity) which should greatly improve the accuracy and utility of the system as proposed.

9.1.2 Contributions

The work undertaken as part of these thesis has provided a greater understanding of the application of deeplearning to a very difficult (and valuable) real world problem. The development of the data set and models used will enable future research in the area to home into more successful final solutions to the problem of forecasting IGT availability in a usable future fashion.

Contributions going forward are the provision of further evidence that a configuration agnostic digital twin can be built.That both the configuration of the engines and there installed equipment can be accurately identified and that they can be quite closely modelled.

9.2 More Data and Training Time

The overall structure of the system could still be tuned for much greater learning capability at a minimal additional cost of computation, however it would require a much bigger and continuous training set in order to be able to pick up on other features that will be present within the existing overall database. Items that were not adequately represented in the current dataset include :

- Compressor waterwash cycles : No continuous operational cycles were knowingly included as the data was extracted on 30 day operational segments
- Annual service cycles : due to the small periods targeted on in relation to operation there were no opportunities to train on the full operational cycle of the engine
- Accumulated hours : Engines will deteriorate in a predictable way based on the hours accumulated, but no information is available to the network in relation to this at the time for it to differentiate on. This could be included in future versions.
- Message Logs : The messages generated by each machine could also be included. Although this was initial investigated it was seen that there would be a significant improvement in visibility of system health as many issues on an IGT do not have continuous measurements recorded, but will have a messages triggered.

There is also evidence (such as the topics touched on regarding the difference in optimisers generability and capability to overcome local minima with sufficient training) that many of the experiments be left to run for longer periods of times without the aggressive use of early stopping for instance. In part this was down to time restrictions due to the length of time required to prepare the data and to iterate through the various design decisions that are detailed within this work.

9.3 Additional Parameters Relating to Environment

Another factor that could be taken into account is the encoding of location data for the various Turbines. This could represent a number of different elements such as Longitude, Latitude and altitude, which all have a bearing on the performance of the engine (due to the changes in climatic performance). However we would also need to identify or classify some of the other potential conditions in the areas such as dust or salt contamination within the inlet air. This could be achieved through clustering of the engines degradation as opposed to the current amalgamated digital twin that has been produced within this current work. Some features are contained within configuration databases as to the assumed environmental conditions, but these have never been validated with the overall performance of the gas turbine. This has been useful in much of the research going into prediction of compressor fouling (such as in [Igie, 2017]).

9.4 Reduction of dimensionality

No significant attempt to reduce the dimensionality of the data reviewed was undertaken in this masters as the intention was to attempt to allow the resulting system appear as end to end as possible.

9.5 Use of AutoML

Given the ability for the simpler digital twin components to differentiate so easily elements such as configuration and tag availability, it possible that more tuned network could be produced using AutoML techniques

9.6 Integration of other problem domains

It has been noted that some of the core representations used within this network should be useful in some of the other domains that have explored the use of Deep Learning and their application to IGT's. There is no reason that some of the core layers could not be shared with the other approaches allowing for greater utility in the field as a system that can both predict availability and diagnose any reasons behind poor availability.

This would be different to the ensemble approach taken in [Osigwe et al., 2017] or [Tayarani-Dathaie and Khorasani, 2015] which are structured very much like a standard computer program (which as should be noted elements of the proposed GAPS solution is currently guilty of in order to enable the solution to be more feasible within the time frame stipulated by this masters programme) with individual models representing discrete sub-routines or fault diagnostic routines. But the central representations contained within the CNN core of the GAPS system is very similar to those presented by [Martinez-Garcia et al., 2019] and [Liu et al., 2018] and could easily have a specialising dealing with such outputs. This would also fit in more closely with the movement towards End-to-End type systems a.k.a Software 2.0.

9.7 Transfer Learning

[Zhong et al., 2019] has proposed used transfer learning for the analysis of bearing faults by utilising more generalised CNN models which have already learned a large number of workable features and the (in this case a network trained on X-Rays) to process the arrays of data produced by sensors.

9.7.1 Generative Approaches

There has been a significant buzz recently in relation to the realm of generative networks that could alleviate some of the issues identified during the research portion of this work that identified that there was serious issues with the lack of available data for the training of networks, and especially that it was often quite seriously unbalanced in relation to not having sufficient populations of examples of systems that have failed. These are well described in [Goodfellow, 2017] and could alleviate considerably the amount of training data that is neccessary as another network could be leverage to produce realistic facsimiles of the engine and conditions that are desired.

9.8 Potential Novel Architectures

During the programme of research that has been undertaken a number of new developments could provide avenues for further development, and expand the scope of the problems that could be tackled.

9.8.1 Temporal Convolutional Memory

[Jayasinghe et al., 2019] proposed a novel layer which combines both a 1D convolutional network with a LSTM to try and learn both the immediate and longer term patterns. This is also then paired with a traditional LSTM to take into account the longer overall sequences and learn the cross sensor patterns. This appears to be a relevant structure in relation to the that which is being considered. LSTM's in general have also been used by a number of parties (such as [Martinsson, 2016; Zhang et al., 2018] and could also be explored further, this is likely to be even more important once a reference point for hours run is available within the dataset.

9.8.2 Capsule Networks

These have been discussed for nearly 20 years, however there have been numerous recent advantage and there is the possibility that they prove useful in the near future. A recent papers by [Sabour et al., 2017] identified a new way of both training and representing data within a neural network through the use of "capsules" which self organise (or dynamically routed) between each others as they become sensitive to certain features within the training problem space. This is seen as potentially being able to overcome some of the problems that occur with spacial understanding that appear to get lost in the common CNN type networks. This may allow for the network to learn some of the implicit spatial relationships that are unique to each individual IGT especially in systems such as combustion where the inherent geometry of it can have a significant effect on its performance but is also to all intents and purposes an unknown quantity. Some of the issues in relation to mapping this will hopefully be minimised by training on a larger dataset, however the capsule network architecture potentially allows a short cut for the network to build up a working representation of the combustion process. As can be seen from the output of the generalised model **??**, there was still an overall shift despite quite closely matching the overall pattern.

9.8.3 Ordinary Differential Equation Networks

[Chen et al., 2019] these are a relatively new and novel network which may provide some significant avenues for the future development of systems such as that described here. There are intended to operate using continuous rather then discrete data points and employ the direct use of calculus in order to achieve that goal. The focus of the research in this case was classification of medical conditions from continuous variables classified against nosy spare categorical information. this may have some advantage when it comes to integrate the message and event logs of the IGT (and potentially even softer forms of information such as the descriptions in fault reports)

9.8.4 Weight Agnostic Neural Network

An interesting approach to developing neural networks is to look at identifying an architecture that is inherently correct in building such an appropriate prediction based on the input data. This is the approach taken by Gaier [Gaier and Ha, 2019]. This

Figure 9.1: Example combustion chamber. Please note that the there is constraints such as probe fitting depth, offset from flame, position and machined accuracy of the nozzles and air guidance pathways that mean that individual burners can have substantially different behaviours, but still behaviour approximately in a similar fashion. Copyright Siemens AG - Used with permission

takes view that the network itself should inherently be capable of answering the question that has been asked without any actual weight optimisation. So far the initial results have been positive in relation to the networks being developed are often simplified compared to those that have been explicitly engineered. They are formed and selected by use of a form of the NEAT application in order to provide the most appropriate network. This evolves the network using a standard evolutionary approach. It would be interesting to identify if that is appropriate for either a sub component or indeed the full system as described within in this work. The promise here is small robust networks, the trade-off however is the increased time in training and growing of the model given it takes an evolutionary approach.

9.9 Conclusions

The author hopes that by this stage the reader has an appreciation for the complexity of the problem at hand, and has understood the achievements gain in relation to undertaking this exercise. In order to conclude this work it is necessary to review the objectives set out in 1.3, and highlight the work to undertaken to meet those.

As can be deduced from the results the project can not yet demonstrate the viability of the approach taken. The investigation and the work of others have shown that this approach appears to be feasible, however as described earlier in this chapter there is still significant further work that needs to be undertaken.

Bibliography

J. Allaire. *TensorFlow for R: Keras for R.* 2017. URL `https://blogs.rstudio.com/tensorflow/posts/2017-09-06-keras-for-r/`.

A. Almasi. Latest lessons learned, modern condition monitoring and advanced predictive maintenance for gas turbines. *Australian Journal of Mechanical Engineering*, 14(3):199–211, 2016.

ALSTOM. THE WORLD‘S FIRST INDUSTRIAL GAS TURBINE SET – GT NEUCHÂTEL : A Historic Mechanical Engineering Landmark, 2007.

N. Batayev and A. Onbayev. Prediction of Gas Turbine Parameters based on Machine Learning Regression Methods. 2018.

R. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud. Neural Ordinary differential Equations. Montreal, Canada, Jan. 2019.

F. Chollet and others. *Keras.* GitHub, 2015. URL `https://github.com/fchollet/keras`.

D.-A. Clevert, T. Unterthiner, and S. Hochreiter. FAST AND ACCURATE DEEP

NETWORK LEARNING BY EXPONENTIAL LINEAR UNITS (ELUS). San Juan, Puerto Rico, Feb. 2016.

S. Cruz-Manzo, V. Panov, and Y. Zhang. Gas Path Fault and Degradation Modelling in Twin-Shaft Gas Turbines. *Machines*, 43(6), 2018. doi: doi:10.3390/machines6040043.

S. Della Villa Jr and C. Koeneke. Monitoring and Measuring the Availability and Reliability Perormanace (RAM) of one OEM's Advanced Gas Turbine Class Fleet: Computation Method Based on Continuous Data Collection. Amersterdam, Netherlands, June 2010. URL `http://www.spsinc.com/News-Events/Technical%20Papers/Monitoring%20and%20Measuring%20the%20Availability%20and%20Reliability%20Performance%20(PowerGen%20Europe%202010).pdf`.

O. Diallo. *A data analytics approahc to gas turbine prognostics and health management*. Ph.D, Georgia Institute of Technology, Dec. 2010.

M. Fast, M. Assadi, and S. De. Development and multi-utility of an ANN model for an industrial gas turbine. *Applied Energy*, (86):9–17, 2009a.

M. Fast, T. Palme, and A. Karlsson. Gas turbine sensor validation through classification with artificial neural networks. In *Proceedings of ECOS 2009*, Foz do Iguacu, Parana, Brazil, Aug. 2009b.

X. Feng, H. Yao, and S. Zhang. An efficient way to refine DenseNet. *Signal, Image and Video Processing*, 13:959–965, 2019.

A. Gaier and D. Ha. Weight Agnostic Neural Networks. 2019.

I. Goodfellow. NIPS 2016 Tutorial : Generative Adversarial Networks, Apr. 2017.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 1st edition, 2016. ISBN 978-0-262-03561-3.

Grandview Research. Gas Turbine Service Market Worth $41.6 Billion By 2025 | CAGR 8.5%, 2018. URL `https://www.grandviewresearch.com/press-release/global-gas-turbine-services-market`.

R. Grosshauser. Turbines vs. Reciprocating Engines. *Power Engineering*, Nov. 2016. URL `https://www.power-eng.com/articles/print/volume-120/issue-11/features/turbines-vs-reciprocating-engines.html`.

R. Hahnloser, R. Sarpeshkar, M. Mahowald, R. Douglas, and S. Seung. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, 405:947–951, June 2000.

D. Hubel. SINGLE UNIT ACTIVITY IN STRIATE CORTEX OF UNRESTRAINED CATS. *The Journal of Phusiology*, 147:226–238, Dec. 1958.

D. Hubel and T. Wiesel. RECEPTIVE FIELDS OF SINGLE NEURONES IN THE CAT'S STRIATE CORTEX. *The Journal of Phusiology*, 148, Apr. 1959.

U. Igie. Gas Turbine Compressor Fouling and Washing in Power and Aerospace Propulsion. *Journal of Engineering for Gas Turbines and Power*, 139, Dec. 2017.

T. Jack. Experience with Industrial Gas Turbine Overhaul: Economy Study. *Journal of Scientific and Engineering Research*, 2(2):71–78, 2015. ISSN 2394-2630.

L. Jayasinghe, T. Samarasinghe, C. Yuen, J. Chen Ni Low, and S. S. Ge. Temporal Convolutional Memory Networks for remaining Useful Life Estimation of Industrial Machinery. Melbourne, Australias, Feb. 2019. doi: doi:10.1109/ICIT.2019.8754956.

F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu. Deep neural networks: A promisingtool for fault characterstic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical Systems and Signal Processing*, 72:303–315, Nov. 2015. doi: doi:10.1016/j.ymssp.2015.10.0025.

R. Kamarudin. *Siemens Gas Turbines Fingerprinting for Maintenance Event Prediction with Machine Learning*. PhD thesis, The University of Sheffield, Sept. 2018.

S. Khan and T. Tairi. A review on the application of deep learning in system health management. *Mechanical Systems and Signal Processing*, (107):242–262, Feb. 2018. doi: doi:10.1016/j.ymssp.2017.11.024.

S. Kiakojoori and K. Khorasani. Dynamic neural networks for gas turbine engine degredation prediction, health monitoring, and prognosis. *Neural Computing & Applications*, 27:2157 –2192, 2016.

Kiefer and Wolfowitz. Stochastic Estimation of the Maximum of a Regression Function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.

D. Kingma and J. Ba. Adam : A method For Stochastic Optimization. San Diego, USA, May 2015.

R. Kurz, K. Brun, and M. Wollie. Degredation Effects on Industrial Gas Turbines. *Journal of Engineering for Gas Turbines and Power*, 131, Nov. 2009. doi: 10.1115/1.3097135.

Y. LeCun, L. Bottou, and P. Haffner. Gradient Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278 – 2324, Nov. 1998.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436, May 2015. URL `https://doi.org/10.1038/nature14539`.

J. Liu, D. Yu, M. Kang, and Y. Weizhong. Fault Detection for Gas Turbine Hot Components Based on a Convolutional Neural Network. *Energies*, 11, 2018.

O. Macmann, K. Cohen, T. Seitz, and A. Behbahani. Performing Diagnositcs & Prognostics on Simulated Engine Failures Using Neural Networks. USA, July 2016. doi: DOI:10.2514/6.2016-4807.

M. Martinez-Garcia, Y. Zhang, J. Wan, and J. McGinty. Visually Interpretable Profile Extraction with an Autoencoder for Health Monitoring of Industrial Systems. Japan, July 2019.

E. Martinsson. *WTTE-RNN : Weibull Time To Event Recurrent Neural Network*. PhD thesis, Chalmers University of Technology, 2016.

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Y. Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. URL `http://tensorflow.org/`.

M. Muhammad, M. Tahan B., A. Karim, and E. Tsoutsanis. Performance-based health monitoring, diagnostics and prognostics for condition-based maintenance of gas turbines: A review. *Applied Energy*, 2017.

E. Osigwe, L. Yi-Guang, S. Sampath, and J. Gbanaibolou. Integrated Gas Turbine System Diagnostics: Components and Sensor Faults Quantification using Artificail Neural Network. Manchester, Sept. 2017.

A. S. Palau, K. Bakliwal, M. H. Dhada, T. Pearce, and A. K. Parlikad. Recurrent Neural Networks for real-time dstributed collaborative prognostics. *IEEE Internal Conference on Prognostics and Health Management*, 2018.

M. Pinelli, P. Spina, and M. Venturini. Gas Turbine Health State Determination:Methodology Approach and Field Application. *International Journal of Rotating Machinery*, 2012, 2012. doi: doi:10.1155/2012/142173.

P. Ramachandran, B. Zoph, and Q. v.Le. Searching for Activation Functions, Oct. 2017.

E. Ramasso and A. Saxens. Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets. *International Journal of Prognostics and Health Management,*, 2014.

S. Sabour, N. Froost, and G. Hinton. Dynamic Routing Between Capsules. Nov. 2017. URL `https://arxiv.org/pdf/1710.09829.pdf`.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout : A Simple Way to Prevent Neural Networks from Over
tting. *Journal of Machine Learning Research*, 15:1929–1958, June 2014.

S. Tayarani-Dathaie and k. Khorasani. Fault detection and isolation of gas turbine engines using a bank of neural networks. *Journal of Process Control*, 36:22–41, 2015.

R. C. Team. R: A Language and Environment for Statistical Computing, 2013. URL `http://www.R-project.org/`.

W. Tiddens, A. J. Braaksma, and T. Tinga. Selecting Suitable Candidates for Predictive Maintenance. *International Journal of Prognostics and Health Management,*, 2018.

M. Venturini and N. Puggina. Predction reliability of a statistical methodology for Gas Turbine Prognostics. *Journal of Engineering for Gas Turbines and Power*, 134, 2012.

T. Wong and Z. Luo. Recurrent Auto-Encoder Model for Large-Scale Industrial Sensor Signal Analysis. *19th International Conference on Engineering Applications of Neural Networks*, 2018.

M. Zaidan, R. Relan, A. Mills, and R. Harrison. Prognostics of gas turbine engine: An integrated approach. *Expert Systems With Applications*, 42:8472 – 8483, 2015.

J. Zhang, P. Wang, R. Yan, and R. Gao. Deep Learning for Imporved System Remaining Life Prediction. Stockholm, Sweden, May 2018. Elsevier.

Y. Zhang, C. Bingham, M. Garlick, and M. Gallimore. Applied Fault Detection and Diagnosis for Industrial Gas Turbine Systems. *International Journal of Automation and Computing*, 14(4):463–473, Aug. 2017. doi: DOI:10.1007/s11633-016-0967-5.

R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. Gao. Deep learning and its applicaitons to machine health monitoring. *Mechanical Systems and Signal Processing*, 115:213–237, 2019.

S.-s. Zhong, S. Fu, and L. Lin. A novel gas turbine fault diagnosis method based on transfer learning with CNN. *Measurement*, (137):435–453, Jan. 2019. doi: doi:10.1016/j.measurement.2019.01.022.